



**TUGAS AKHIR - TE 145561**

**PERANCANGAN *SOFT STARTING* PADA MOTOR  
INDUKSI 3 FASA MENGGUNAKAN MIKROKONTROLER  
ATMEGA328**

Yoki Rijal Fauzi  
NRP 2214038025

Dosen Pembimbing  
Ir. Arif Musthofa, MT.  
Suwito, ST., MT.

PROGRAM STUDI TEKNIK LISTRIK  
Departemen Teknik Elektro Otomasi  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

-----Halaman ini sengaja dikosongkan-----



**FINAL PROJECT - TE 145561**

**DESIGN SOFT STARTING FOR 3 PHASE MOTOR INDUCTION  
USING MICROCONTROLLER ATMEGA328**

Yoki Rijal Fauzi  
NRP 2214038025

Advisor I  
Ir. Arif Musthofa, MT.

Advisor II  
Suwito, ST., MT.

ELECTRICAL ENGINEERING STUDY PROGRAM  
Electrical and Automation Engineering Department  
Vocational Faculty  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

-----Halaman ini sengaja dikosongkan-----

## PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul "**Perancangan Soft Starting pada Motor Induksi 3 Fasa Menggunakan Mikrokontroler Atmega328**" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 20 Juli 2017

Mahasiswa



Yoki Rijal Fauzi  
NRP 2214038025

-----Halaman ini sengaja dikosongkan-----

**PERANCANGAN *SOFT STARTING* PADA MOTOR INDUKSI  
3 FASA MENGGUNAKAN MIKROKONTROLER ATMEGA328**

**TUGAS AKHIR**

Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Ahli Madya  
Pada

Program Studi Teknik Listrik  
Departemen Teknik Elektro Otomasi  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember

Menyetujui:

Dosen Pembimbing I

Dosen Pembimbing II

Ir. Arif Musthofa, MT.  
NIP. 196608 11199203 1 004

Suwito, ST., MT.  
NIP. 19810105 200501 1 004

**SURABAYA  
JULI, 2017**

---Halaman ini sengaja dikosongkan---



# **PERANCANGAN *SOFT STARTING* PADA MOTOR INDUKSI 3 FASA MENGGUNAKAN MIKROKONTOLER ATMEGA328**

**Nama : Yoki Rijal Fauzi**  
**Pembimbing 1 : Ir. Arif Musthofa, MT.**  
**Pembimbing 2: Suwito, ST., MT.**

## **ABSTRAK**

Motor Induksi mempunyai banyak keunggulan di segi teknis maupun ekonomis, karena itu motor induksi terutama jenis motor induksi 3 fasa banyak digunakan pada dunia industri. Akan tetapi motor induksi juga mempunyai kekurangan, antara lain arus awal (*start*) yang besar yang dapat mengakibatkan penurunan tegangan sistem dan mengganggu kerja sistem peralatan lain dalam satu saluran. Maka dari itu diperlukan metode *Soft Starting* untuk mengurangi lonjakan arus pada arus awal (*start*) yang besar. Metode *Soft Starting* dilakukan dengan menggunakan komponen elektronika daya yaitu triac. Tegangan masukan motor induksi diatur dengan memberikan sinyal tunda pada triac dengan waktu tertentu yang diatur oleh *platform* Arduino. Pada beban motor induksi 3 fasa, besar arus *starting* relatif terkendali dan lebih kecil jika dibandingkan dengan metode DOL dan *Wye-Delta*.

Hasil dari perancangan Modul Starting Motor Induksi 3 Fasa dengan metode *Soft Starting* mempunyai pemilihan waktu *soft start* antara 4 detik hingga 25 detik, sehingga dalam rentang waktu tersebut tegangan input yang masuk bertahap mulai 1% hingga 100% dari tegangan 380v *Delta*.

**Kata Kunci :** *Soft Starting*, Penyulutan Triac, *Starting* Motor Induksi 3 Fasa.

-----Halaman ini sengaja dikosongkan-----

## ***DESIGN SOFT STARTING FOR 3 PHASE MOTOR INDUCTION USING MICROCONTROLLER ATMEGA328***

**Name : Yoki Rijal Fauzi**  
**Advisor 1 : Ir. Arif Musthofa, MT.**  
**Advisor 2 : Suwito, ST., MT.**

### ***ABSTRACT***

*Induction motor has many advantages in terms of technical and economical, therefore induction motors, especially types of 3-phase induction motors are widely used in the industrial world. However, induction motor also has disadvantages, such as the starting of large currents (start) which may lead to voltage drop system and interfere with the work of other equipment systems in the same line. Therefore, Soft Starting method is needed to reduce the current spikes on the starting motor induction. Soft Starting method is done by using power electronics component that is triac. The input voltage of the induction motor is regulated by providing a delay signal on the triac with a specified time set by the Arduino platform. At the 3 phase induction motor load, the starting current is relatively controllable and smaller then the DOL and Wye-Delta methods.*

*The result of the design 3 Phase Induction Motor Starting Module with Soft Starting method has soft start time selection between 4 second until 25 second, so in that time period input voltage entering gradually from 1% to 100% from 380v Delta voltage*

***Keywords : Soft Starting, Triac, Starting 3 Phase Motor Induction.***

-----Halaman ini sengaja dikosongkan-----

## KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT atas dua nikmat yang sangat berharga yaitu nikmat sehat dan juga nikmat waktu sehingga Tugas Akhir ini dapat terselesaikan dengan baik. Shalawat serta salam semoga selalu dilimpahkan kepada Rasulullah Muhammad SAW, keluarga, sahabat, dan umat muslim yang senantiasa meneladani beliau.

Tugas Akhir ini disusun untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan Diploma-3 pada Bidang Studi Teknik Teknik Listrik, Departemen Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember Surabaya dengan judul:

### **PERANCANGAN *SOFT STARTING* PADA MOTOR INDUKSI 3 FASA MENGGUNAKAN MIKROKONTOLER ATMEGA328**

Dalam Tugas Akhir ini dirancang sebuah metode *Soft Starting* dengan komponen utama yaitu *triac* yang berbasis mikrokontroler ATmega328 dengan *platform* Arduino untuk dilakukan pengasutan pada motor induksi tiga fasa.

Penulis mengucapkan terima kasih kepada Ibu dan Bapak penulis yang memberikan berbagai bentuk doa serta dukungan tulus tiada henti, Bapak Ir. Arif Musthofa, MT. dan Bapak Suwito S.T., MT. atas segala bimbingan ilmu, moral, dan spiritual dari awal hingga terselesaikannya Tugas Akhir ini, Angkyn Ilmawati yang selalu memberikan doa, semangat, dan dukungannya kepada penulis. Penulis juga mengucapkan banyak terima kasih kepada teman teman kelas D3 PLN secara umumnya dan warga Kene secara khususnya, tentunya semua pihak yang telah membantu dalam proses penyelesaian Tugas Akhir ini.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Tugas Akhir ini. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya, 20 Juli 2017



Penulis

-----Halaman ini sengaja dikosongkan-----

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN JUDUL .....	iii
PERNYATAAN KEASLIAN TUGAS AKHIR .....	v
HALAMAN PENGESAHAN .....	vii
ABSTRAK .....	ix
<i>ABSTRACT</i> .....	xi
KATA PENGANTAR .....	xiii
DAFTAR ISI .....	xv
DAFTAR GAMBAR .....	xvii
DAFTAR TABEL .....	xix

### BAB I PENDAHULUAN

1.1 Latar Belakang .....	1
1.2 Permasalahan .....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan .....	3
1.5 Metodologi Penelitian .....	3
1.6 Sistematika Laporan .....	3
1.7 Relevansi .....	4

### BAB II TEORI DASAR

2.1 Motor Listrik .....	5
2.2 Prinsip Kerja Motor Listrik .....	5
2.3 Motor Induksi Tiga Fasa .....	6
2.3.1 Klasifikasi Motor Induksi .....	6
2.3.2 Prinsip Kerja Motor Induksi 3 fasa .....	6
2.4 Metode-Metode Starting Motor Tiga Fasa .....	8
2.4.1 Starting dengan Metode Direct On Line .....	9
2.4.2 Starting dengan Metode Autotransformer .....	10
2.4.3 Starting dengan Metode Star-Delta .....	11
2.4.4 Starting dengan Metode Soft Starter .....	12
2.5 Komponen-Komponen Elektronika Daya .....	13
2.5.1 Dioda .....	13
2.5.2 Transistor .....	16
2.5.3 Thyristor .....	16
2.6 Mikrokontroler ATmega328 .....	17
2.7 Zero Crossing Detector .....	19

2.8 MOC3021 .....	20
<b>BAB III PERANCANGAN ALAT</b>	
3.1 Blok Diagram Seluruh Sistem <i>Soft Starting</i> .....	23
3.2 Prinsip Kerja Alat .....	25
3.3 Spesifikasi Motor Induksi 3 Fasa .....	26
3.4 Perancangan Hardware .....	27
3.4.1 Perancangan Rangkaian Zero Cross Detector ...	27
3.4.2 Perancangan Rangkaian Solid State Relay .....	28
3.4.3 Perancangan Rangkaian Interface .....	29
3.5 Perancangan Software .....	30
3.6 Perancangan Box Alat .....	32
<b>BAB IV PENGUJIAN DAN ANALISA DATA</b>	
4.1 Pengujian Arduino Uno .....	35
4.2 Pengujian <i>Zero Cross Detector</i> .....	38
4.3 Pengujian Sinyal PWM .....	40
4.4 Pengujian Rangkaian Daya .....	44
4.5 Pengujian Sistem Keseluruhan .....	51
<b>BAB V PENUTUP</b>	
5.1 Kesimpulan .....	63
5.2 Saran .....	63
<b>DAFTAR PUSTAKA</b> .....	65
<b>LAMPIRAN A</b>	
A.1. <i>Listing</i> Program pada Arduino .....	A-1
<b>LAMPIRAN B</b>	
B.1 Datasheet Arduino Uno .....	B-1
B.2 Datasheet PC817 .....	B-5
B.3 Datasheet MOC3021 .....	B-9
B.4 Datasheet BT138 .....	B-13
<b>LAMPIRAN C</b>	
C.1 Tampilan <i>Board</i> .....	C-1
C.2 Tampilan Depan Modul .....	C-2
C.3 Tampilan Didalam Modul .....	C-2
<b>DAFTAR RIWAYAT HIDUP</b> .....	D-1



## DAFTAR GAMBAR

<b>Gambar 2.1</b> Motor Listrik.....	5
<b>Gambar 2.2</b> Prinsip Kerja Motor Listrik.....	6
<b>Gambar 2.3</b> Motor tipe rotor sangkar tupai( <i>Squirrel-cage rotor</i> ).....	7
<b>Gambar 2.4</b> Diagram <i>Direct On Line Starter</i> .....	9
<b>Gambar 2.5</b> Diagram <i>Autotransformer Starter</i> .....	11
<b>Gambar 2.6</b> Diagram <i>Star-Delta Starter</i> .....	12
<b>Gambar 2.7</b> Diagram <i>Soft Starter</i> .....	13
<b>Gambar 2.8</b> Persambungan P-N .....	14
<b>Gambar 2.9</b> Pembiasan Dioda .....	14
<b>Gambar 2.10</b> Dioda Bridge.....	15
<b>Gambar 2.11.</b> Dioda Zener .....	15
<b>Gambar 2.12.</b> Simbol dan Gambar Transistor .....	16
<b>Gambar 2.13.</b> Simbol dan Struktur Triac.....	17
<b>Gambar 2.14</b> Mikrokontroler ATmega328 <i>pin mapping</i> .....	18
<b>Gambar 2.15</b> Arduino Uno dengan <i>chip</i> ATmega328 .....	19
<b>Gambar 2.16</b> Sinyal <i>Zero Crossing Detector</i> .....	20
<b>Gambar 2.17</b> Simbol MOC3021 .....	21
<b>Gambar 3.1</b> Rangkaian <i>Full Soft Starter</i> dengan TRIAC .....	24
<b>Gambar 3.2</b> Blok Diagram <i>Soft Starter</i> dengan TRIAC .....	26
<b>Gambar 3.3</b> Name Plate Motor .....	26
<b>Gambar 3.4</b> Motor Induksi Tiga Fasa .....	27
<b>Gambar 3.5</b> Skematik <i>Zero Cross Detector</i> .....	28
<b>Gambar 3.6</b> Skematik <i>Solid State Relay</i> .....	29
<b>Gambar 3.7</b> Skematik Rangkaian <i>Interface</i> .....	30
<b>Gambar 3.8</b> <i>Flowchart</i> Sistem <i>Soft Starting</i> .....	31
<b>Gambar 3.9</b> Perancangan Box Tampak Sampling .....	32
<b>Gambar 3.10</b> Perancangan Box Tampak Depan .....	33
<b>Gambar 3.11</b> Perancangan Box Tampak Atas .....	33
<b>Gambar 3.12</b> Perancangan Box Tampak Belakang .....	34
<b>Gambar 3.13</b> Perancangan Desain Akrilik.....	34
<b>Gambar 4.1</b> <i>Flowchart</i> Pengujian Arduino .....	36
<b>Gambar 4.2</b> Pengujian Arduino. ....	36
<b>Gambar 4.4</b> Rangkaian <i>Zero Cross Detector</i> pada Tanda Kuning.....	39
<b>Gambar 4.5</b> Pengujian <i>Zero Cross Detector</i> dengan Osiloskop pada Fasa T .....	39
<b>Gambar 4.6</b> Sinyal <i>Input</i> Fasa T-N dan <i>Output Zero Cross</i> <i>Detector</i> .....	40

<b>Gambar 4.7</b>	<i>Flowchart</i> Sinyal PWM.....	41
<b>Gambar 4.8</b>	Pengujian Sinyal PWM dengan Osiloskop pada Pin 11 Arduino .....	42
<b>Gambar 4.9</b>	Gelombang PWM 30% pada Tahap ke 3.....	43
<b>Gambar 4.10</b>	Gelombang PWM 60% pada Tahap ke 6.....	44
<b>Gambar 4.12</b>	Rangkaian Daya dengan Tanda Warna Merah .....	45
<b>Gambar 4.13</b>	Pengujian Rangkaian Daya dengan Osiloskop . ...	46
<b>Gambar 4.14</b>	Gelombang Penyulutan <i>Triac</i> Bertegangan 55 V ..	46
<b>Gambar 4.15</b>	Gelombang Penyulutan <i>Triac</i> Bertegangan 175 V	47
<b>Gambar 4.16</b>	Gelombang Penyulutan <i>Triac</i> Bertegangan 206 V	47
<b>Gambar 4.17</b>	Gelombang Penyulutan <i>Triac</i> Bertegangan 225 V	48
<b>Gambar 4.18</b>	Penyalan Lampu Secara <i>Soft Start</i> pada 0.1 s .....	49
<b>Gambar 4.19</b>	Penyalan Lampu Secara <i>Soft Start</i> pada 8 s .....	50
<b>Gambar 4.20</b>	Penyalan Lampu Secara <i>Soft Start</i> pada 15s .....	50
<b>Gambar 4.21</b>	<i>Wiring Soft Starting</i> Motor Induksi Tiga Fasa.....	52
<b>Gambar 4.22</b>	<i>Wiring Soft Starting</i> Motor Induksi Tiga Fasa.....	53
<b>Gambar 4.23</b>	Menu <i>Home</i> pada LCD .....	53
<b>Gambar 4.24</b>	Empat Menu <i>Display</i> pada LCD .....	54
<b>Gambar 4.25</b>	Tampilan Menu “ <i>Set Soft Start</i> ”.....	54
<b>Gambar 4.26</b>	Tampilan Menu “ <i>Set Soft Start</i> ” Setelah <i>Set Time</i>	55
<b>Gambar 4.27</b>	Tampilan Menu “ <i>Soft Starting</i> ” .....	55
<b>Gambar 4.28</b>	Tampilan Menu “ <i>Soft Starting</i> ” Penyulutan Maksimal .....	56
<b>Gambar 4.29</b>	Grafik Arus <i>Soft Starting</i> 10 Detik .....	57
<b>Gambar 4.30</b>	Grafik Arus <i>Soft Starting</i> 15 Detik .....	58
<b>Gambar 4.31</b>	Grafik Arus <i>Soft Starting</i> 20 Detik .....	59
<b>Gambar 4.32</b>	Grafik Arus <i>Soft Starting</i> 25 Detik .....	60

## DAFTAR TABEL

<b>Tabel 4.1</b> Hasil Pengujian Arduino Uno.....	37
<b>Tabel 4.2</b> Hasil Pengukuran Sumber Tiga Fasa <i>Line to Line</i> .....	38
<b>Tabel 4.3</b> Hasil Pengukuran Waktu Penyulutan <i>Triac</i> Tanpa Beban .....	48
<b>Tabel 4.4</b> Hasil Pengukuran Waktu Penyulutan <i>Triac</i> dengan Lampu .....	49
<b>Tabel 4.5</b> Hasil Pengukuran Penyulutan <i>Triac</i> Line R dengan Line T .....	51
<b>Tabel 4.6</b> <i>Soft Starting</i> Dengan Rentang Waktu 10 Detik .....	56
<b>Tabel 4.7</b> <i>Soft Starting</i> Dengan Rentang Waktu 15 Detik .....	57
<b>Tabel 4.8</b> <i>Soft Starting</i> Dengan Rentang Waktu 20 Detik .....	58
<b>Tabel 4.9</b> <i>Soft Starting</i> Dengan Rentang Waktu 25 Detik .....	59

*Halaman ini sengaja dikosongkan*

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Perkembangan teknologi di masa sekarang sangat pesat pertumbuhannya itu terlihat dari semakin banyaknya industri-industri yang berdiri dan memproduksi alat-alat teknologi masa kini. Motor-motor induksi sangat penting penggunaannya sebagai alat bantu penggerak peralatan lain seperti pada industri perakitan alat-alat transportasi, perakitan alat berat maupun pada industri pertambangan motor induksi sangat dibutuhkan, paling banyak digunakan dalam industri dengan skala besar maupun kecil dan di dalam rumah tangga alasannya adalah bahwa karakteristiknya hampir sesuai dengan kebutuhan dunia industri, pada umumnya dalam kaitannya dengan harga, kesempurnaan, pemeliharaan, dan kestabilan kecepatan.

Motor induksi (*asinkron*) ini pada umumnya hanya memiliki satu suplai tenaga yang mengeksitasi belitan stator, belitan rotornya tidak terhubung langsung dengan sumber tenaga listrik, melainkan belitan ini dieksitasi oleh induksi dari perubahan medan magnetik yang disebabkan oleh arus pada belitan stator. Hampir semua motor AC yang digunakan adalah motor induksi, terutama motor induksi tiga fasa yang paling banyak dipakai di perindustrian.

Motor induksi tiga fasa sangat banyak dipakai sebagai penggerak di perindustrian karena banyak memiliki keuntungan, tetapi ada juga kelemahannya. Keuntungan motor induksi tiga fasa yaitu kokoh dan kuat, murah dan dapat diandalkan, efisiensi yang tinggi pada keadaan kerja normal, dan perawatannya mudah, sedangkan kelemahannya berupa arus pengasutan awal yang mencapai lima hingga tujuh kali dari arus nominal kerja motor beban penuh, terutama untuk motor berdaya besar.

Arus pengasutan awal yang besar dapat mengakibatkan penurunan tegangan sistem dan mengganggu kerja sistem peralatan lain dalam satu saluran, seperti peralatan-peralatan elektronik, sehingga diperlukan suatu metode pengasutan dengan pengurangan tegangan pada motor induksi yang bertujuan untuk mengurangi arus pengasutan awal.

*Soft Starting* adalah suatu cara penurunan tegangan *starting* dari motor induksi AC. *Soft Starting* merupakan metode *starting* yang prinsipnya sama dengan *starting* motor menggunakan *primary resistance* yang diseri dengan *supply* tegangan ke motor, arus masuk dalam *stator* sama dengan arus keluar. *Soft Starting* terdiri dari

komponen *thyristor* untuk mengontrol aliran arus yang masuk ke motor, sehingga tegangan akan masuk secara bertahap dan akhirnya penuh.

*Soft Starting* bertujuan untuk mendapatkan *start* yang terkendali, sehalus mungkin serta terproteksi dan mencapai kecepatan nominal yang konstan sehingga mendapatkan arus *starting* rendah. Diharapkan tegangan dan arus dari sumber tenaga dapat mengalir masuk kedalam motor AC secara bertahap, sehingga tidak memerlukan arus pengasutan yang besar.

## **1.2 Rumusan Masalah**

Rumusan masalah yang akan dikaji dalam Tugas Akhir ini adalah tidak adanya modul pembelajaran membahas tentang metode *soft starting* motor induksi tiga fasa di Laboratorium Elektronika Terapan pada Teknik Elektro Otomasi. Oleh karena itu dengan modul rangkaian *soft starting* dapat menambah pengetahuan tentang metode *soft starting* pada motor induksi tiga fasa.

## **1.3 Batasan Masalah**

Agar penulisan buku Tugas Akhir ini tidak menyimpang dan mengambang dari tujuan yang semula direncanakan sehingga mempermudah mendapatkan data dan informasi yang diperlukan, maka penulis menetapkan batasan-batasan masalah sebagai berikut:

1. Analisa dibatasi pada respon motor induksi 3 fasa terhadap *soft starter* pada keadaan motor tanpa beban.
2. Spesifikasi motor listrik yang dapat digunakan hanya berkisar 1 HP hingga 2 HP.
3. Tidak dibahas mengenai adanya harmonisa yang ditimbulkan pada saat pengasutan *Soft Starting* motor induksi.
4. Mikrokontroler Arduino Uno yang digunakan untuk mengatur sudut picu pada *triac*.

## **1.4 Tujuan Perencanaan**

Adapun tujuan dari Tugas Akhir ini yakni merancang modul perangkat keras dan perangkat lunak rangkaian *Soft Starting* motor induksi tiga fasa yang menggunakan sebuah mikrokontroler ATmega 328 sebagai kontrol pemecutan sudut picu tegangan. Yang nantinya modul *Soft Starting* motor induksi tiga fasa bisa digunakan sebagai media pembelajaran pada laboratorium Teknik Elektro Otomasi.

## **1.5 Metodologi Penelitian**

Penelitian ini dilakukan melalui beberapa tahapan metodologi, yaitu studi literatur, permodelan sistem, perancangan dan pembuatan alat, analisa data dan penyusunan laporan buku Tugas Akhir.

Pada tahap studi literatur akan dipelajari tentang perangkat dan karakteristik dari macam macam pengasutan motor induksi tiga fasa, mempelajari mikrokontroler Arduino uno, mempelajari macam macam komponen dari elektronika daya yang digunakan sebagai *Soft Starting*, mempelajari rangkaian penyalan sudut fasa menggunakan *triac*. Pada tahap permodelan sistem, dirancang permodelan atau desain dari rangkaian *zero cross detection* serta rangkaian *solid state relay*. Kemudian perancangan software berupa algoritma yang digunakan untuk kontrol *triac* sebagai pengendali tegangan input untuk motor induksi tiga fasa. Yang terakhir adalah merancang hasil permodelan sistem dan pengecekan terhadap sistem secara keseluruhan sehingga diperoleh data pengukuran yang akurat. Data percobaan yang diperoleh selanjutnya akan dianalisa. Dari hasil analisa, akan ditarik kesimpulan dari penelitian yang telah dilakukan. Tahapan akhir penelitian adalah penyusunan laporan penelitian.

## **1.6 Sistematika Laporan Tugas Akhir**

Sistematika pembahasan Tugas Akhir ini terdiri dari lima bab, yaitu pendahuluan, teori penunjang, perencanaan dan pembuatan alat, pengujian dan analisa alat, serta penutup.

### **Bab I : PENDAHULUAN**

Pada bab ini membahas tentang latar belakang permasalahan, permasalahan, tujuan, sistematika laporan, serta relevansi.

### **Bab II : TEORI PENUNJANG**

Pada bab ini membahas tentang teori penunjang yang mendukung dalam perencanaan pembuatan alat meliputi definisi dan prinsip kerja motor induksi tiga fasa, teori macam macam pengasutan motor induksi tiga fasa, teori definisi dari komponen komponen elektronika daya yang digunakan sebagai perancangan *hardware Soft Starting*.

- Bab III : PERENCANAAN DAN PEMBUATAN ALAT**  
Membahas tentang perencanaan dan pembuatan perangkat keras (*hardware*) yang meliputi perancangan rangkaian *zero cross detector*, rangkaian *solid state relay*, rangkaian mikrokontroler. Serta perangkat lunak (*software*) berupa program untuk mengatur sudut penyalan *triac* berdasarkan pilihan rentang waktu yang diinginkan.
- Bab IV : PENGUJIAN DAN ANALISA ALAT**  
Membahas tentang pengukuran, pengujian, dan penganalisaan terhadap komponen-komponen fisik seperti pengukuran tegangan pada *input* maupun *output*, pengujian *zero cross detector*, pengujian rangkaian daya kendali tegangan AC, pengukuran dan pengujian metode *Soft Starting*.
- Bab V : PENUTUP**  
Menjelaskan tentang kesimpulan dari tugas akhir ini dan saran-saran untuk pengembangan alat ini lebih lanjut.

### 1.7 Relevansi

Dengan adanya tugas akhir ini diharapkan dapat dijadikan sebagai tambahan materi Praktikum pada laboratorium D3 Teknik Elektro ITS dan juga salah satu solusi media pembelajaran bagi mahasiswa atau pelajar supaya dapat lebih memahami tentang beberapa macam starting motor 3 fasa, sehingga kedepannya dapat diaplikasikan pada dunia industri.



## **BAB II**

### **TEORI DASAR**

Pada bab ini membahas tentang teori dasar dan teori penunjang dari peralatan-peralatan yang digunakan dalam pembuatan Tugas Akhir “Perancangan *Soft Starting* pada Motor Induksi 3 Fasa Menggunakan Mikrokontroler ATmega328”.

#### **2.1 Motor Listrik**

Motor listrik adalah alat untuk mengubah energi listrik menjadi energi mekanik. Alat yang berfungsi sebaliknya, mengubah energi mekanik menjadi energi listrik disebut generator atau dinamo. Motor listrik dapat ditemukan pada peralatan rumah tangga seperti kipas angin, mesin cuci, pompa air dan penyedot debu. Motor listrik yang umum digunakan di dunia Industri adalah motor listrik asinkron, dengan dua standar global yakni IEC dan NEMA. Motor asinkron IEC berbasis metrik (milimeter), sedangkan motor listrik NEMA berbasis imperial (inch), dalam aplikasi ada satuan daya dalam horsepower (hp) maupun kiloWatt (kW).

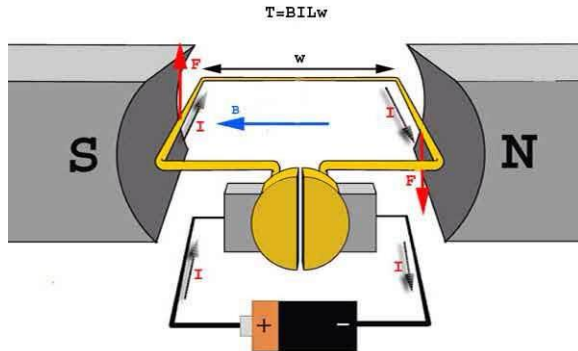


**Gambar 2.1 Motor Listrik**

#### **2.2 Prinsip Kerja Motor Listrik**

Pada motor listrik tenaga listrik diubah menjadi tenaga mekanik. Perubahan ini dilakukan dengan mengubah tenaga listrik menjadi

magnet yang disebut sebagai elektro magnet. Sebagaimana kita ketahui bahwa : kutub-kutub dari magnet yang senama akan tolak-menolak dan kutub-kutub tidak senama, tarik-menarik. Maka kita dapat memperoleh gerakan jika kita menempatkan sebuah magnet pada sebuah poros yang dapat berputar, dan magnet yang lain pada suatu kedudukan yang tetap. Dapat kita ketahui dari gambar 2.2.

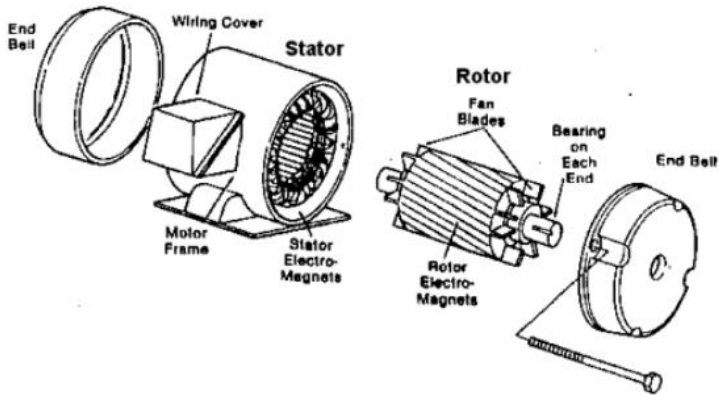


**Gambar 2.2** Prinsip Kerja Motor Listrik

### 2.3 Motor Induksi Tiga Fasa

Motor induksi tiga fasa memiliki dua komponen dasar yaitu stator dan rotor, bagian rotor dipisahkan dengan bagian stator oleh celah udara (*air gap*). Tipe dari motor induksi berdasarkan pada jenis rotor dibagi menjadi dua macam yaitu *Wound* rotor (tipe motor yang memiliki rotor terbuat dari lilitan) dan *Squirrel-cage* rotor (tipe motor induksi dimana konstruksi rotor disusun oleh beberapa batangan logam yang dimasukkan melewati slot-slot yang ada pada rotor motor induksi, kemudian pada setiap bagian disatukan oleh cincin yang membuat batangan logam dihubungkan singkat dengan batangan logam yang lain).

komputer dengan menggunakan kabel USB atau sumber tegangan berasal dari adaptor AC-DC atau baterai untuk menghidupkan arduino .



**Gambar 2.3** Motor tipe rotor sangkar tupai (*Squirrel-cage rotor*)

### 2.3.1 Klasifikasi Motor Induksi

Motor induksi satu fasa. Motor ini hanya memiliki satu gulungan stator, beroperasi dengan pasokan daya satu fasa, memiliki sebuah rotor sangkar tupai, dan memerlukan sebuah alat untuk menghidupkan motornya. Sejauh ini jenis motor ini merupakan jenis motor yang paling umum digunakan dalam peralatan rumah tangga, seperti kipas angin, mesin cuci, dan pengering pakaian dan untuk penggunaan 3 hingga 4 Hp.

Motor induksi tiga fasa. Medan magnet yang berputar dihasilkan dari pasokan sumber tiga fasa yang seimbang. Motor tersebut memiliki kemampuan daya yang tinggi, dapat memiliki kandang tupai atau gulungan rotor (walaupun 90% memiliki rotor kandang tupai), dan penyalan sendiri. Diperkirakan bahwa sekitar 70% motor di industri menggunakan jenis motor ini. Sebagai contoh, pompa, Kompresor, *belt conveyor*, jaringan listrik, dan *grinder*. Tersedia dalam ukuran 1/3 sampai ratusan Hp.

### 2.3.2 Prinsip Kerja Motor Induksi 3 fasa

Motor induksi merupakan motor arus bolak-balik (AC) yang paling luas digunakan. Penamaannya berasal dari kenyataan bahwa arus rotor motor ini bukan diperoleh dari sumber tertentu, tetapi merupakan arus yang terinduksi sebagai akibat adanya perbedaan relatif antara putaran rotor dengan medan putar (*Rotating Magnetic field*) yang dihasilkan oleh arus stator.

Apabila ketiga belitan stator diberikan masing-masing diberi tegangan dari sumber tiga fasa, maka akan timbul medan magnet yang berputar dengan kecepatan sinkron.

$$n_s = \frac{120f}{p} \quad (2.1)$$

dengan:

$n_s$  = Slip motor

$f$  = Frekuensi jaringan ( Hz)

$p$  = Jumlah kutub

Medan putar tersebut akan memotong konduktor rotor hingga terbangkit tegangan induksi. Karena konduktor rotor dihubung singkat, maka akan mengalir arus dalam konduktor rotor. Arus rotor ini berada dalam medan magnet dari stator menurut hukum *Lorenz*, akibatnya timbul gaya/torka. Bila gaya ini cukup untuk menggerakkan rotor maka ia akan berputar dengan kecepatan

$$n_r = (1 - s) n_s \quad (2.2)$$

dengan:

$n_r$  = Kecepatan putar rotor ( rpm)

$s$  = Slip putaran

karena tegangan induksi hanya akan terbangkitkan jika terjadi perpotongan antar medan putar dengan konduktor rotor maka kecepatan rotor tidak dapat menyamai kecepatan medan putar stator, harus ada selisih dimana kecepatan rotor ( $n_r$ ) harus lebih rendah dari kecepatan medan putar (kecepatan sinkron  $n_s$ ). Perbedaan kecepatan ini disebut *slip* ( $s$ ) dan dinyatakan dengan:

$$s = \frac{n_s - n_r}{n_s} \times 100\% \quad (2.3)$$

adanya perbedaan kecepatan medan putar dan rotor ini sehingga mesinnya disebut mesin tak sinkron/serempak.

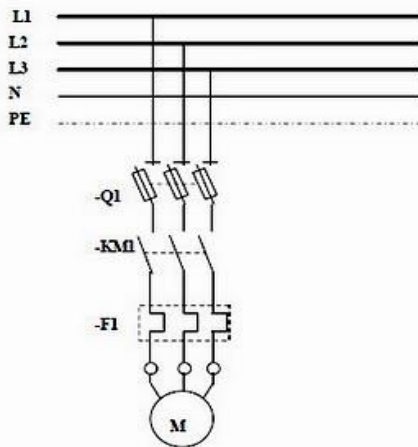
## 2.4 Metode-Metode *Starting* Motor Tiga Fasa

Pada motor induksi yang diam apabila tegangan normal diberikan ke stator, maka akan ditarik arus yang besar oleh belitan primernya. Motor induksi saat dihubung langsung akan menarik arus 5 sampai 7 kali dari arus beban penuh dan hanya menghasilkan torsi 1,5 dan 2,5 kali torsi beban penuh. Arus mula yang besar ini dapat mengakibatkan drop

tegangan pada saluran, sehingga akan mengganggu pada peralatan lain yang dihubungkan pada saluran yang sama. Untuk mengurangi besarnya arus *starting* pada motor induksi, ada beberapa cara atau metode starting yang biasa digunakan.

#### 2.4.1 Starting dengan Metode *Direct On Line*

*Direct On Line* merupakan *starting* langsung. Penggunaan metoda ini sering dilakukan untuk motor-motor AC yang mempunyai kapasitas daya yang kecil. Pengertian penyambungan langsung disini, motor yang akan dijalankan langsung di *switch* on ke sumber tegangan jala-jala sesuai dengan besar tegangan nominal motor. Artinya tidak perlu mengatur atau menurunkan tegangan pada saat *starting*, arus *starting* sama dengan arus hubung singkat. Rangkaian *starting* DOL ini dapat dilihat pada Gambar 2.4



**Gambar 2.4** Diagram *Direct On Line* Starter

Starter ini terdiri dari *Breaker* sebagai proteksi hubung singkat, *Magnetik Contactor*, *Over Current Relay* dan komponen control seperti *push button*, MCB dan *pilot lamp*. Kontrol Start dan Stop dilakukan dengan *push button* yang mengontrol tegangan pada *coil* *contactor*. Sementara itu output OCR terangkai secara seri sehingga jika OCR trip, maka output OCR akan melepas tegangan ke *coil* *contactor*. Komponen penyusun starter ini harus mempunyai *ampacity*

yang cukup besar. Perlu diperhitungkan juga arus saat start motor, demikian juga ukuran *range overload*nya. Keuntungan pengasutan DOL adalah sebagai berikut :

1. Pengasutannya sederhana
2. Jarang salah menjalankannya
3. Instalasinya ekonomis
4. Efisiensi kerjanya tinggi

Kerugian pengasutan DOL adalah sebagai berikut

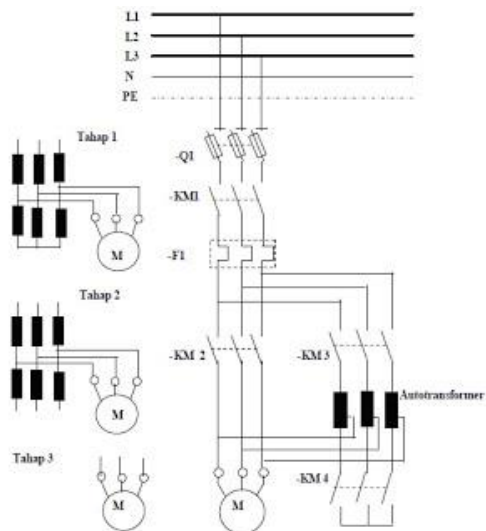
1. Tegangan sumber menurun tiba-tiba
2. Arus asut besar mengakibatkan kopel besar
3. Bila pengasutan sering dilakukan akan membangkitkan panas yang tinggi.

#### **2.4.2 Starting dengan Metode *Autotransformer***

Pengasutan tipe ini mempunyai keuntungan utama yaitu diperoleh harga momen per-unit yang besar dari arus pengusutannya. Besar arus motor berkurang sebanding dengan tegangan terminal motor. Sedangkan arus jala-jala berkurang sebanding dengan arus kuadrat dari tegangan terminal motor, karena pengaruh dari autotrafo. Tetapi biaya pengasutan dengan metode ini umumnya lebih tinggi dibandingkan dengan metode pengasutan lainnya. Autotrafo umumnya dilengkapi dengan dua atau tiga tap, misalnya 80 %, 60 % dan 50 % tap tegangan. Pengasutan dengan autotrafo mengakibatkan ketidakseimbangan tegangan antara fasa. Kumparan trafo hanya digunakan pada saat pengasutan dan setelah itu dapat dilepaskan dari rangkaian. Pelepasan kumparan trafo ini dapat dilakukan secara otomatis dengan menggunakan kontak-kontak rele induksi dan dilengkapi dengan rele waktu. Dibawah ini ditunjukkan gambar pengasutan dengan autotrafo yang menggunakan dua kumparan trafo. Keuntungan pengasutan auto trafo:

1. Reaktansi bocor yang lebih rendah
2. Arus yang dihasilkan lebih kecil
3. Rugi-rugi yang lebih rendah

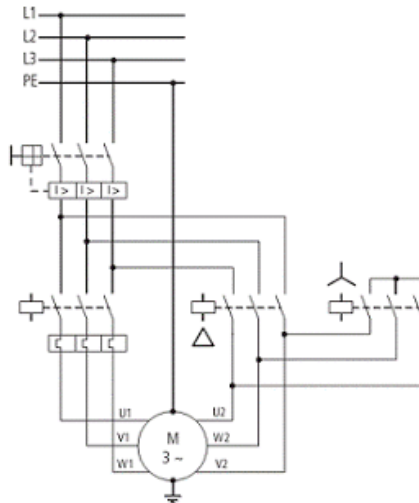
Kerugian pengasutan auto trafo adalah Langsungnya hubungan listrik antara sisi bertegangan tinggi dan rendah. Diagram starting *autotransformer* dapat dilihat pada gambar 2.5



**Gambar 2.5** Diagram *Autotransformer Starter*

### 2.4.3 Starting dengan Metode *Star-Delta*

Pengasutan dengan metode ini digunakan terutama untuk motor induksi rotor sangkar yang direncanakan bekerja/berputar pada hubungan delta. Ujung-ujung kumparan fasanya harus dikeluarkan ke terminal motor, yaitu untuk melakukan hubungan *Wye* dan hubungan *Delta*. Pada tipe ini mula-mula kumparan dihubungkan secara *Wye*, dimana tegangan fasanya adalah sebesar 0.577 dari tegangan kumparan pada saat dihubungkan *delta*, sedangkan impedensi kumparan tersebut bila dihubungkan *delta*. Dengan demikian, arus pengasutan pada hubungan bintang adalah lebih kecil bila dihubungkan dengan *delta*. Demikian juga momen pengasutannya adalah sebanding dengan besarnya arus pengasutan yang timbul. Umumnya alat pengasutan ini dilengkapi dengan kontaktor-kontaktor yang dimiliki oleh masing-masing hubungan tersebut. Perpindahan hubungan *Wye* ke *Delta* didalam sistem pengasutan ini diatur oleh suatu rele pengatur waktu (*Time Delay Relay*). Pada saat perpindahan hubungan *Wye* ke *Delta* terjadi pemutusan arus sesaat dari arus jala-jala. Namun hal ini dapat diatasi dengan menggunakan sistem peralihan tertutup dengan hubungan sebuah tahanan ke dalam rangkaian selama perpindahan berlangsung. Diagram starting *star-delta* dapat dilihat pada gambar 2.6



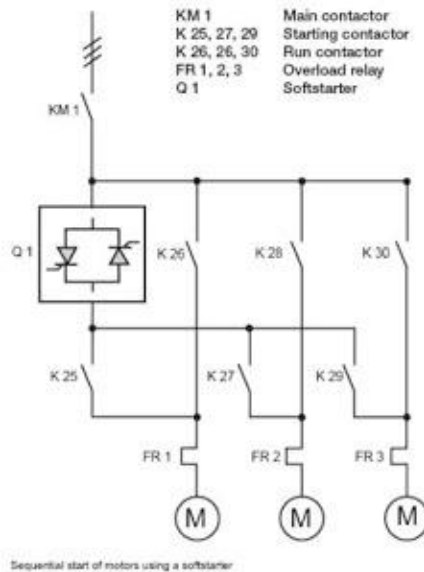
**Gambar 2.6** Diagram *Star-Delta Starter*

#### 2.4.4 Starting dengan Metode *Soft Starter*

*Soft starter* dipergunakan untuk mengatur/ memperhalus start dari elektrik motor. Prinsip kerjanya adalah dengan mengatur tegangan yang masuk ke motor. Pertama-tama motor hanya diberikan tegangan yang rendah sehingga arus dan torsi pun juga rendah. Pada level ini motor hanya sekedar bergerak perlahan dan tidak menimbulkan kejutan. Selanjutnya tegangan akan dinaikan secara bertahap sampai ke nominal tegangannya dan motor akan berputar dengan kondisi RPM yang nominal.

Komponen utama softstarter adalah thyristor dan rangkaian yang mengatur *trigger thyristor*. Seperti diketahui, *output thyristor* dapat di atur via pin *gate* nya. Rangkaian tersebut akan mengontrol level tegangan yang akan dikeluarkan oleh *thyristor*. *Thyristor* yang terpasang bisa pada 2 phase atau 3 phase. Diagram rangkaian metode *soft starter* dapat dilihat pada gambar.2.7





**Gambar 2.7** Diagram *Soft Starter*

## 2.5 Komponen-Komponen Elektronika Daya

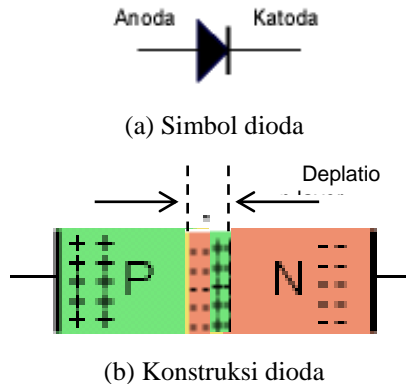
Dalam dunia elektronika, selain kita mengenal komponen elektronika aktif dan pasif, juga terdapat salah satu jenis komponen lainnya, yakni komponen elektronika daya. Pasti sebagian dari anda banyak yang bertanya-tanya sebenarnya apa itu elektronika daya.

Perlu diketahui bahwa sebenarnya elektronika daya adalah cabang ilmu elektronika yang membahas terkait switching, pengontrolan serta pengubah atau konversi blok blok yang besar dari daya listrik menggunakan sarana alat berbahan semikonduktor. Secara garis besar, elektronika terbagi menjadi dua macam, yakni rangkaian daya serta rangkaian kontrol.

### 2.5.1 Dioda

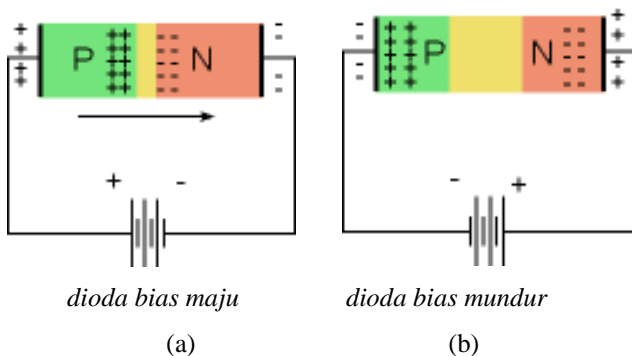
Dioda merupakan komponen elektronika yang hanya bisa mengalirkan arus listrik satu arah saja, yaitu dari anoda ke katoda, hal ini disebabkan adanya potensial penghalang (*Potential Barrier*) pada persambungan PN. Material pembentuk dioda adalah semi konduktor

dari bahan silikon atau germanium tipe P dan N, seperti terlihat pada gambar 2.8.



**Gambar 2.8** Persambungan P-N

Struktur dalam dioda adalah sambungan semikonduktor tipe P dan N, yang mengakibatkan munculnya lapisan deplesi pada persambungannya. Dengan struktur demikian arus hanya akan dapata mengalir dari sisi P menuju sisi N, jika pada anoda dan katodanya diberikan tegangan maju seperti pada gambar 2.9 a, Jika anoda dan katodanya diberikan tegangan mundur seperti pada gambar 2.9 b, maka tidak terjadi aliran arus pada dioda. Sifat khusus dari dioda inilah yang kemudian melandasi terbentuknya konsep rangkaian penyearah arus AC menjadi DC.



**Gambar 2.9** Pembiasan Dioda

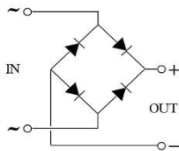
Ada beberapa jenis dioda seperti pada berikut :

1. Dioda Generic

Dioda generic yang paling umum digunakan sebagai penyearah tegangan AC dari jala-jala listrik ke tegangan DC yang siap dikonsumsi untuk berbagai keperluan catu daya. Proses penyearahan AC ke DC

2. Dioda Bridge

Secara bahan, fungsi dan karakteristik dioda bridge sama dengan dioda generic, bedanya hanya pada dalam satu kemasan dioda bridge tersusun dari empat dioda generic, seperti terlihat pada gambar 3.6.



(a) Simbol

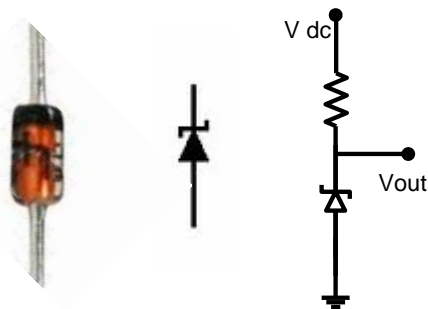


(b) Jenis

**Gambar 2.10** Dioda Bridge

3. Dioda Zener

Dioda zener merupakan jenis dioda yang memanfaatkan kondisi *breakdown*, sehingga pemasangan dioda ini dalam rangkaian, dipasang secara *reverse* untuk mendapatkan titik PIV (*Peak Inverse Voltage*) yang telah ditentukan oleh pabrik. Dengan adanya PIV pada dioda zener ini, maka dimungkinkan pembatasan level tegangan yang stabil dalam sebuah sistem rangkaian elektronika . Contoh rangkaian dapat dilihat pada gambar 2.11.

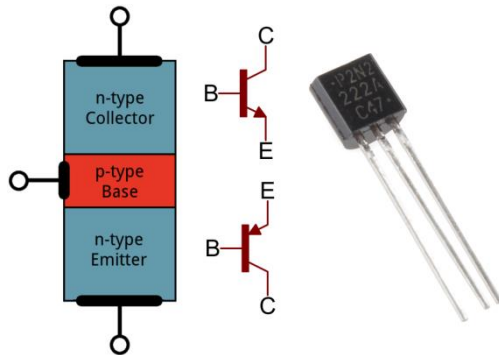


(a) Rangkaian dasar dioda zener

**Gambar 2.11.** Dioda Zener

### 2.5.2 Transistor

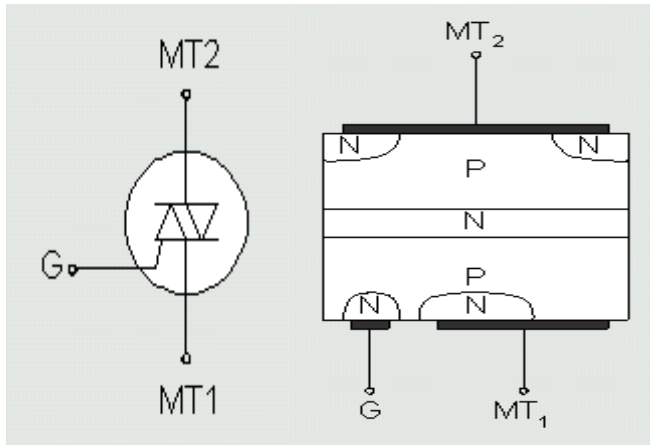
Transistor adalah komponen elektronika dasar yang juga masuk ke dalam jajaran komponen elektronika aktif. Komponen yang memiliki fungsi sebagai penguat dan switch ini juga termasuk salah satu jenis komponen elektronika daya yang sering dijumpai di berbagai jenis rangkaian elektronika.



**Gambar 2.12.** Simbol dan Gambar Transistor

### 2.5.3 Thyristor

Thyristor berakar kata dari bahasa Yunani yang berarti ‘pintu’. Dinamakan demikian karena sifat dari komponen yang mirip dengan pintu yang dapat dibuka dan ditutup untuk melewatkan arus listrik. Ciri - ciri utama dari sebuah thyristor adalah komponen yang terbuat dari bahan *semiconductor silicon*. Thyristor adalah komponen semikonduktor untuk pensaklaran yang berdasarkan pada struktur PNPN. Komponen ini memiliki kestabilan dalam dua keadaan yaitu on dan off serta memiliki umpan-balik regenerasi internal. Thyristor memiliki kemampuan untuk mensaklar arus searah (DC) yaitu jenis SCR, maupun arus bolak-balik (AC), jenis TRIAC. contoh thyristor yaitu triac pada gambar 2.13



**Gambar 2.13.** Simbol dan Struktur Triac

## 2.6 Mikrokontroler ATmega328

Mikrokontroler ATmega328 atau pada jaman sekarang bisa disebut dengan Arduino Uno sebagai *platformnya* secara sederhana, adalah chip yang menjadi otak dari sebuah rangkaian elektronika. Analoginya, jika pada komputer anda otak utama adalah *chip processor* Intel Core i7 (atau AMD) yang mengatur semua *hardware* dari mulai *motherboard*, RAM, *hardisk*, *monitor*, *keyboard*, dsb, maka pada rangkaian mikrokontroler Arduino, chip ATmega328 adalah otaknya, yang mengatur komponen lain yang terhubung dengan nya (misal led, sensor, motor). Jika pada komputer and *software* diinstall dengan *Operating System Windows* (atau Mac OSX / Linux) sebagai tuan rumah nya, pada mikrokontroler *software* berbentuk program yang langsung di 'bakar' (*burn*) langsung pada *chip* mikrokontroler yang bersangkutan, dengan '*bootloader*' sebagai tuan rumahnya. Pada platform Arduino sekarang, chip yang populer digunakan adalah ATmega328. Platform Arduino, yang dibantu oleh dengan board-board Arduino, memudahkan anda melakukan memrograman chip ATmega328 untuk melakukan tugas yang anda tentukan.

ATmega328 adalah chip mikrokontroler 8-bit berbasis AVR-RISC buatan Atmel. *Chip* ini memiliki 32 KB memori ISP flash dengan kemampuan baca-tulis (*read write*), 1 KB EEPROM, dan 2 KB SRAM. Dari kapasitas memori Flash nya yang sebesar 32 KB itulah chip ini

diberi nama ATmega328. Chip lain yang memiliki memori 8 KB diberi nama ATmega8, dan ATmega16 untuk yang memiliki memori 16 KB.

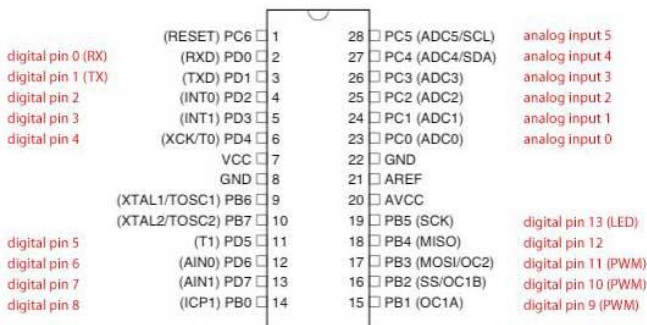
Chip ATmega328 memiliki banyak fasilitas dan kemewahan untuk sebuah chip mikrokontroler. Chip tersebut memiliki 23 jalur general purpose I/O (input/output), 32 buah register, 3 buah timer/counter dengan mode perbandingan, interrupt internal dan external, serial programmable USART, 2-wire interface serial, serial port SPI, 6 buah channel 10-bit A/D converter, programmable watchdog timer dengan oscillator internal, dan lima power saving mode. Chip bekerja pada tegangan antara 1.8V ~ 5.5V. Output komputasi bisa mencapai 1 MIPS per Mhz. *Maximum operating frequency* adalah 20 Mhz.

ATmega328 menjadi cukup populer setelah chip ini dipergunakan dalam board Arduino. Dengan adanya Arduino yang didukung oleh *software* Arduino IDE, pemrograman *chip* ATmega328 menjadi jauh lebih sederhana dan mudah.

Pada dasarnya *board* Arduino adalah tentang ATmega328. Board Arduino didesain untuk mempermudah kita untuk melakukan pemrograman dan koneksi *chip* ATmega328 dengan komponen lainnya. Dengan *board* Arduino, baik itu Arduino UNO, Arduino Mega 2560, Arduino Nano, maupun Arduino Pro Mini, semua nya membantu untuk menyederhanakan proses membuat rangkaian mikrokontroler. Contoh *pin mapping* dari mikrokontroler pada gambar 2.14 dan gambar arduino uno pada gambar 2.15

Arduino Pin Mapping

[www.arduino.cc](http://www.arduino.cc)



Gambar 2.14 Mikrokontroler ATmega328 Pin Mapping



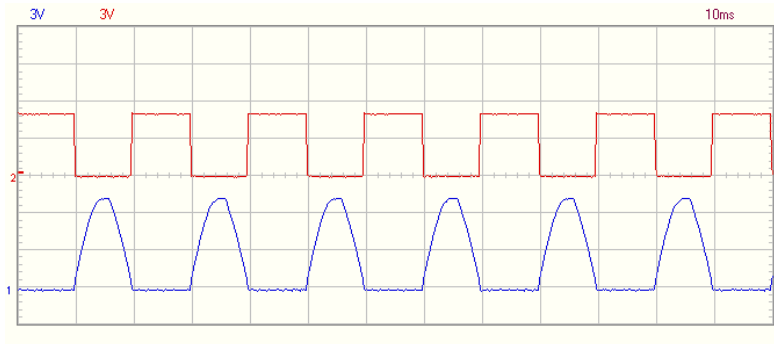
**Gambar 2.15** Arduino Uno dengan *Chip* ATmega328

## **2.7 Zero Crossing Detector**

*Zero crossing detector* adalah rangkaian yang digunakan untuk mendeteksi gelombang sinus AC 220 volt saat melewati titik tegangan nol. Seberangan titik nol yang dideteksi adalah peralihan dari positif menuju negatif dan peralihan dari negatif menuju positif. Seberangan-seberangan titik nol ini merupakan acuan yang digunakan sebagai awal pemberian nilai waktu tunda untuk pemicuan *triac*.

Metode ini berfungsi untuk menentukan frekuensi suatu gelombang dengan cara mendeteksi banyaknya zero point pada suatu rentang waktu. *Zero crossing detector* berfungsi untuk mendeteksi perpotongan gelombang sinus pada tegangan AC dengan *zero point* tegangan AC tersebut, sehingga dapat memberikan sinyal acuan saat dimulainya pemicuan sinyal PWM. Dengan menggunakan rangkaian *zero crossing detector* ini, kita dapat mendeteksi *zero point* sekaligus mengubah suatu sinyal sinusoidal (*sine wave*) menjadi sinyal kotak (*square wave*). Perpotongan titik nol yang terdeteksi adalah pada saat peralihan dari siklus positif menuju siklus negatif dan peralihan dari

siklus negatif menuju siklus positif. Contoh sinyal ada pada gambar 2.16 warna merah adalah output berupa sinyal kotak, warna biru adalah sinyal input.



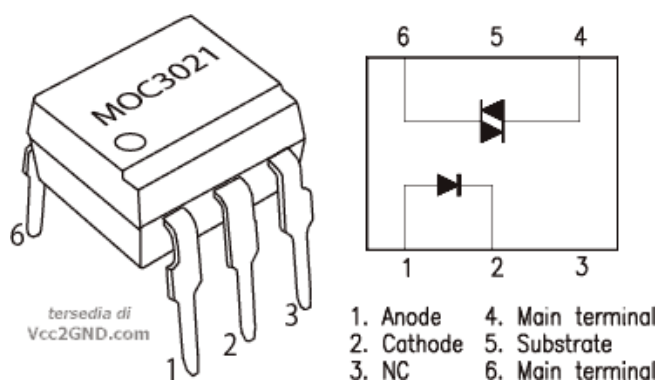
**Gambar 2.16** Sinyal Zero Crossing Detector

## 2.8 MOC3021

MOC3021 adalah Optocoupler / Optoisolator TRIAC Driver yang dirancang untuk mengarah-kendalikan rangkaian elektronika bertegangan 220 Volt AC (listrik PLN) dalam kemasan standar DIP 6-pin.

Aplikasinya meliputi pengendali katup solenoid, *ballast* lampu, sebagai antarmuka antara MCU / rangkaian bertegangan rendah searah (*low voltage DC circuit*) dengan peralatan arus bolak-balik (*alternating current*) bertegangan hingga 240 Volt AC, pengarah-kendali motor elektrik (*AC motor driver*), sebagai pengatur intensitas cahaya lampu bohlam (*incandescent lamp dimmer*), dsb. Komponen elektronika saklar ganda (*bilateral switch*) ini menawarkan isolasi hingga 7500 Volt (kondisi puncak). Sehingga dengan komponen ini dapat melindungi mikrokontroler dari tegangan 220 volt AC. Contoh MOC3021 pada gambar 2.17





**Gambar 2.17** Simbol MOC3021

*Halaman ini sengaja dikosongkan*

## BAB III

### PERANCANGAN ALAT

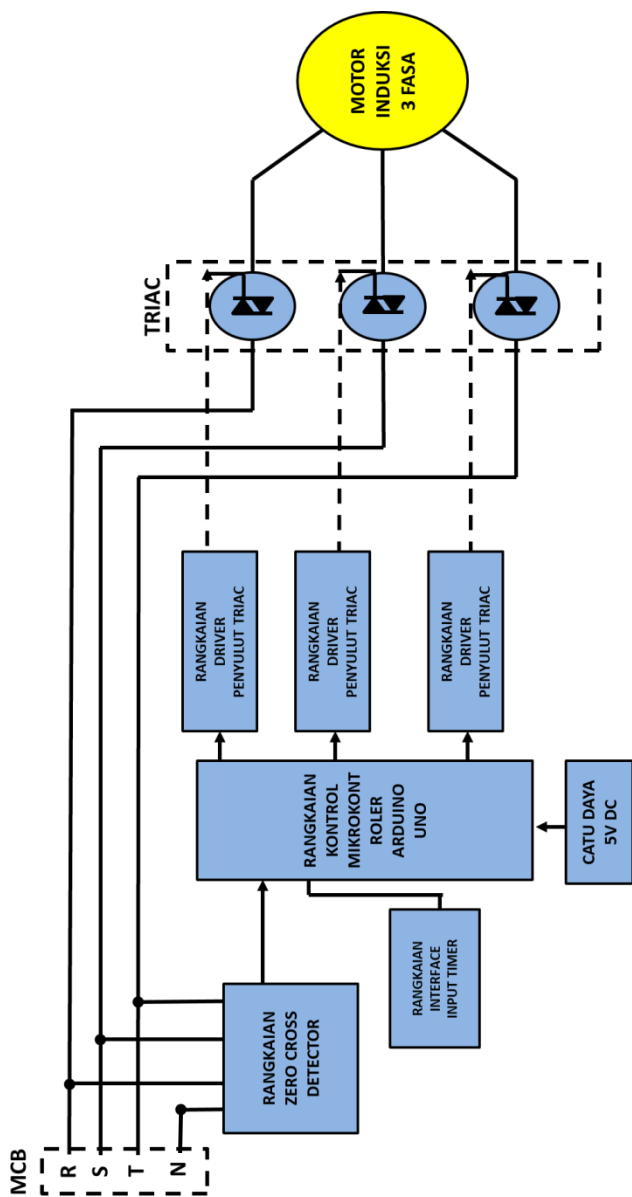
Pada bab ini dibahas mengenai perancangan “Perancangan *Soft Starting* pada Motor Induksi 3 Fasa Menggunakan Mikrokontroler ATmega328”, baik perancangan perangkat elektronika (hardware) serta perangkat lunak atau pemrograman (software) yang meliputi :

- a. Blok diagram seluruh sistem dari alat yang akan dirancang
- b. Menentukan spesifikasi motor induksi 3 fasa
- c. Perancangan Hardware, terdiri dari :
  - Perancangan *Zero Cross Detector*,
  - Perancangan *Solid State Relay*, dan
  - Perancangan *Interface*
- d. Perancangan Software yang berupa flowchart
- e. Perancangan Box alat

#### 3.1 Blok Diagram Seluruh Sistem *Soft Starting*

Perancangan perangkat keras (*hardware*) pada *Soft Starting* meliputi rangkaian catu daya 5V DC, deteksi fasa (*zero cross detector*), sistem minimum ATmega328, rangkaian pemisah dan penyulut Thyristor (*Solid State Relay*), rangkaian *Soft Starting* dan rangkaian *interface*. Dengan adanya perancangan ini diharapkan hasil dari pembuatan alat akan sesuai dengan yang diharapkan. Proses pembuatan alat yaitu rancang bangun *soft starter* untuk motor induksi 3 fasa 0.25kW berbasis mikrokontroler ATmega 328 yang dibuat dari mulai penentuan spesifikasi komponen, langkah-langkah perancangan dan langkah-langkah mengoperasikan alat.

Tujuan dari perancangan alat ini menghasilkan alat yang berfungsi untuk memperhalus arus start motor induksi 3 fasa rotor sangkar dengan kapasitas daya output 0.25kW. Dan menghasilkan alat untuk sistem pengasutan dengan metode *soft starter* yang digunakan untuk motor induksi 3 fasa rotor sangkar dengan kapasitas daya output 0.25kW yang dioperasikan secara otomatis. Gambar blok diagram dari seluruh sistem perancangan *Soft Starting* dapat dilihat pada gambar 3.1 berikut.



**Gambar 3.1** Rangkaian *Full Soft Starter* dengan TRIAC

Rangkaian *Soft Starting* dihubungkan secara seri dengan saluran tegangan yang masuk ke stator motor, *Soft Starting* bertujuan untuk mendapatkan proses pengasutan yang terkendali. Rangkaian *Soft Starting* motor induksi 3 fasa memiliki spesifikasi sebagai berikut:

1. Sumber tegangan yang digunakan adalah AC tiga fasa 380V, 50Hz yang merupakan tegangan masukan rangkaian daya *Soft Starting*.
2. Kemampuan hantar arus maksimal rangkaian daya *Soft Starting* tiga fasa sebesar 10A.
3. Kemampuan tegangan maksimal rangkaian daya *Soft Starting* tiga fasa sebesar 500V.
4. Rangkaian kontrol untuk *Soft Starting* tiga fasa menggunakan sumber tegangan masukan AC satu fasa 220V, 50Hz.
5. Rangkaian *Soft Starting* tiga fasa terdiri dari satu buah mikrokontroler yang masing – masing mengatur satu tegangan fasa dan bekerja secara bersamaan.
6. mikrokontroler mendeteksi tiga fasa dan mengatur proses pemutusan tiga buah *thyristor* yang bekerja secara bergantian.
7. Terdapat 3 tombol masukan, yaitu tombol *Up*, *Down*, *Menu* dan terdapat tiga masukan deteksi fasa.

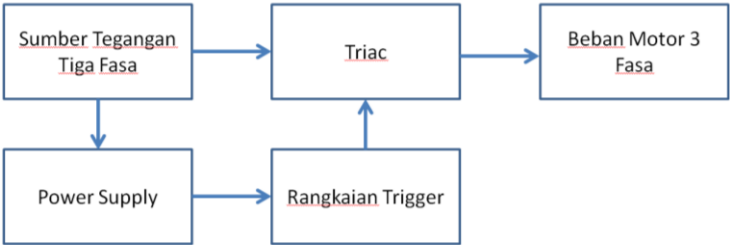
### 3.2 Prinsip Kerja Alat

Prinsip kerja dari metode *soft starter* ini adalah dengan mengatur tegangan yang akan masuk ke motor. Pertama-tama motor akan diberi tegangan yang rendah sehingga arus dan torsi-nya rendah. Pada kejadian ini motor hanya bergerak perlahan dan tidak menimbulkan kejutan. Selanjutnya tegangan dinaikan tahap demi tahap sampai pada tegangan nominalnya dan motor akan berputar dengan kondisi putaran yang nominal.

Pada perencanaannya waktu yang dibutuhkan untuk proses *soft starter* ini yaitu dibutuhkan waktu satu detik untuk motor bisa bekerja pada keadaan nominalnya. Dari waktu satu detik ini dicacah lagi menjadi 10 tahap waktu untuk proses *Soft Starting* dengan cara memperpendek waktu untuk penyulutan sudut  $\alpha$  pada gelombang tegangan dengan tujuan agar motor menerima tegangan dari sumber sedikit demi sedikit.

Pada dasarnya untuk mengendalikan suatu motor listrik dibutuhkan suatu rangkaian kontrol dan rangkaian daya. Begitu juga pada sistem pengasutan motor induksi tiga fasa rotor sangkar 0.25 kW dengan metoda *soft starter* yang akan dibuat ini membutuhkan rangkaian

kontrol, rangkaian daya dan komponen pendukung yang lainnya. Pada bagian rangkaian daya, untuk mencacah tegangan dibutuhkan suatu komponen elektronika daya yaitu TRIAC. Perhatikan gambar 3.2 yang merupakan blok diagram pada sistem kerja alat *Soft Starting* pada motor induksi 3 fasa.



**Gambar 3.2** Blok Diagram *Soft Starter* dengan TRIAC

### 3.3 Spesifikasi Motor Induksi 3 Fasa

Kami menggunakan motor induksi 3 fasa sebagai beban pada alat kami. Berikut pada gambar 3.3 adalah spesifikasi motor induksi 3 fasa yang kami gunakan.



**Gambar 3.3** Name Plate Motor

Dari gambar 3.3 didapatkan spesifikasi Motor tiga fasa sebagai berikut :

1. Daya : 0.25 kW
2. Frekuensi : 50 Hz
3. D *kopling* : 380 Volt
4. *I starting* : 1.4 A
5. *I steady state* : 0.8 A
6. Kecepatan putar : 1340 rpm

Berikut pada gambar 3.4 bentuk dari motor induksi 3 fasa yang kami gunakan sebagai beban. Pada bagian atas motor kami buat agar terdapat *female* untuk *banana jack*.



**Gambar 3.4** Motor Induksi Tiga Fasa

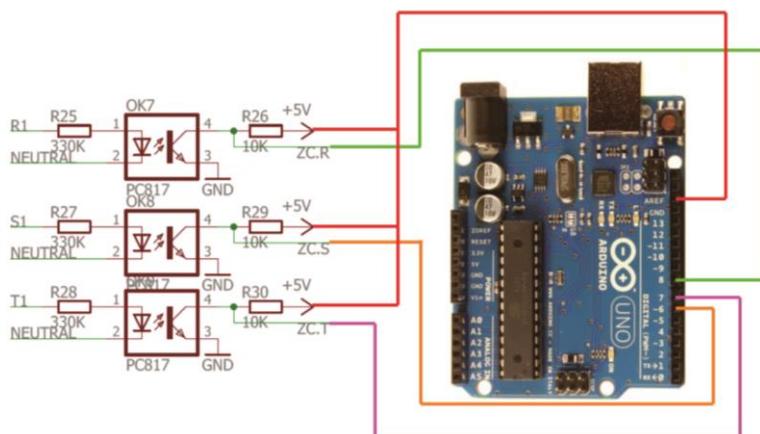
### **3.4 Perancangan *Hardware***

Perancangan *hardware* ini terdiri dari kumpulan beberapa fungsi rangkaian komponen yang membentuk suatu sistem yang berfungsi untuk penyulutan pada TRIAC. Rangkaian dan fungsi yang ada pada alat yang akan dirancang ini diantaranya yaitu :

#### **3.4.1 Perancangan Rangkaian *Zero Cross Detector***

Pada tugas akhir ini *zero cross detector* digunakan untuk pendeteksi nol pada gelombang sinus soida yang dimanfaatkan sebagai

input *firing angle* pada mikrokontroler ATmega328. Terdapat tiga buah rangkaian *zero cross detector* yang masing masing terhubung dengan salah satu fasa R, S dan T. Ketiganya menggunakan optocoupler tipe PC817 dan dua buah resistor sebesar 10 k $\Omega$  dan 330 k $\Omega$ . Output dari optocoupler yang bersumber dari fasa R terhubung dengan pin D8 Arduino Uno, untuk output dari *optocoupler* yang bersumber dari fasa S terhubung dengan pin D7 Arduino Uno, sedangkan output dari *optocoupler* yang bersumber dari fasa T terhubung dengan pin D6 Arduino Uno. Skematik dari masing masing rangkaian *zero cross detector* tersebut seperti ditampilkan pada Gambar 3.5



**Gambar 3.5** Skematik *Zero Cross Detector*

### 3.4.2 Perancangan Rangkaian *Solid State Relay*

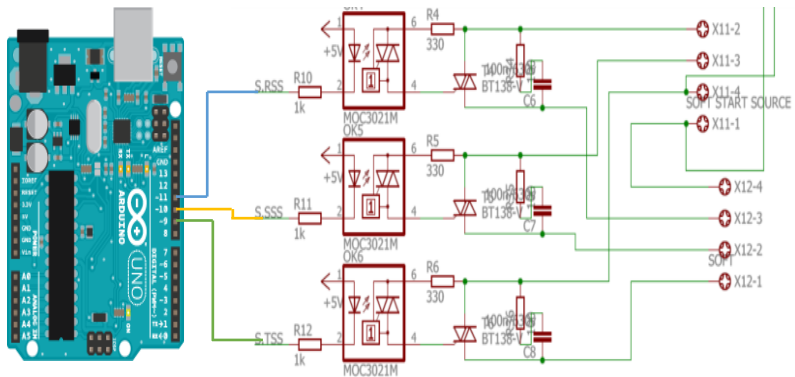
Rangkaian *Solid State Relay* ini berfungsi sebagai rangkaian isolasi yang memisahkan antara rangkaian control dan rangkaian daya. Apabila terjadi gangguan pada rangkaian daya, maka rangkaian solid state relay akan melindungi bagian kontrolnya atau melindungi ATmega328. Sehingga pada mikrokontroler ATmega328 bisa diamankan oleh rangkaian ini. Komponen yang digunakan pada rangkaian ini adalah :

1. *Optocoupler* MOC3021
2. Resistor 1k ohm
3. Resistor 330 ohm



Berikut pada gambar 3.6 untuk susunan rangkaian *solid state relay*.

Berikut pada gambar 3.6 untuk susunan rangkaian *solid state relay*.



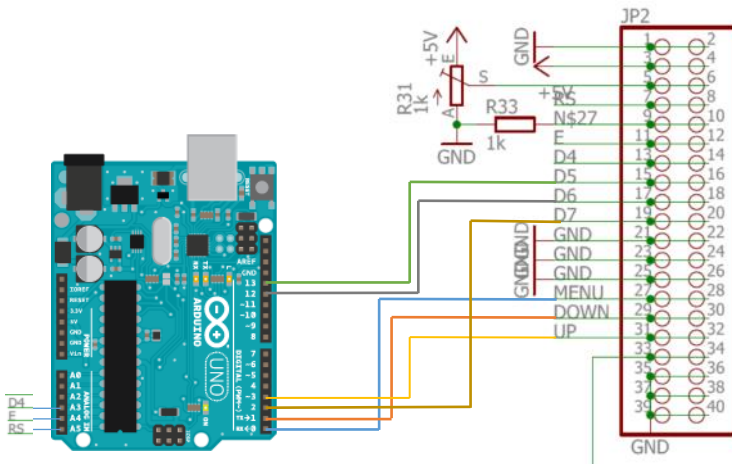
### Gambar 3.6 Skematik *Solid State Relay*

Dapat diketahui dari gambar 3.6 Rangkaian *solid state relay* terdapat *thyristor* BT138 yang merupakan komponen utama untuk dilakukannya *Soft Starting* pada motor tiga fasa yang dipisahkan dengan *optocoupler* MOC3021 untuk pengaman dari tegangan tinggi ke mikrokontroler.

### 3.4.3 Perancangan Rangkaian *Interface*

Rangkaian *Interface* ini berfungsi sebagai input timer yang diinginkan yang kemudian diinputkan kedalam mikrokontroler ATmega328 lalu diproses menjadi rentang waktu untuk proses sudut penyalan tegangan pada *triac*. rangkaian ini terdiri dari :

1. LCD
2. Tombol
3. Resistor 33 ohm
4. Resistor variable



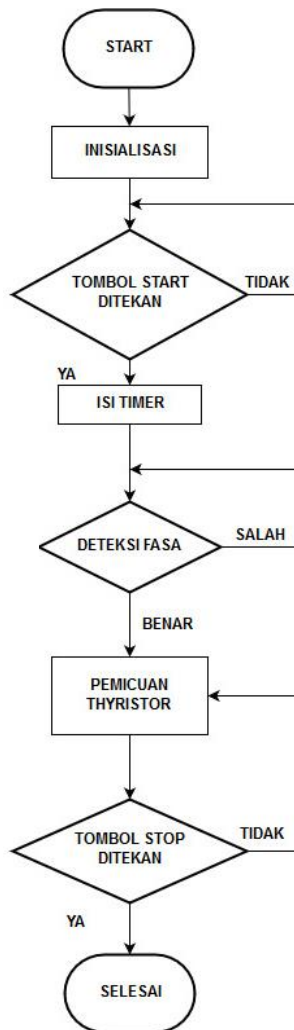
**Gambar 3.7** Skematik Rangkaian *Interface*

Tombol-tombol yang dimaksudkan pada rangkaian diatas adalah tombol *up*, *down* dan *menu*. Dimana tombol-tombol tersebut berguna untuk input timer berupa angka masuk ke mikrokontroler yang kemudian diproses untuk rentang waktu dilaksanakannya *Soft Starting*. Dimana tampilan input dari tombol-tombol *up*, *down* dan *menu* ditampilkan pada LCD 4 x 16 yang diatur kontrasnya dengan resistor variable

### 3.5 Perancangan Software

Perancangan perangkat lunak dikemukakan mengenai perancangan program perintah untuk mikrokontroler. Berupa instruksi – instruksi deteksi fasa dan *Soft Starting*, dimana programnya diisikan ke dalam mikrokontroler Atmega 328 atau Arduino Uno. Untuk mempermudah dalam melakukan pemrograman pada mikrokontroler, terlebih dahulu dibuat diagram alir (*flowchart*) dari perancangan program. Berikut *flowchart* pada gambar 3.8

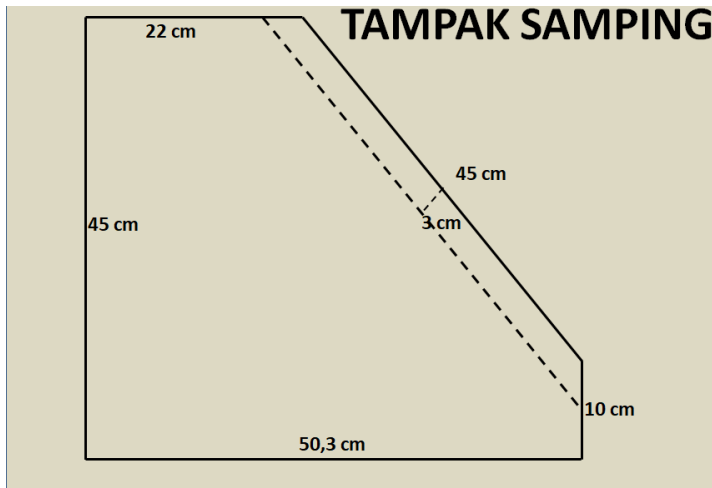
Dari *flowchart* gambar 3.8, dibuat program dalam bahasa *assembly*, setelah itu di *compile* menggunakan *software* ArduinoIDE yang kemudian diisikan pada satu buah mikrokontroler yang berfungsi untuk mengatur seluruh proses *Soft Starting*.



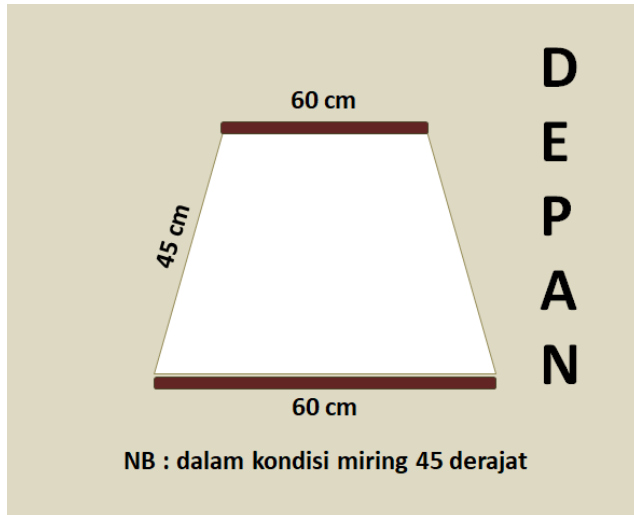
**Gambar 3.8** *Flowchart Sistem Soft Starting*

### 3.6 Perancangan Box Alat

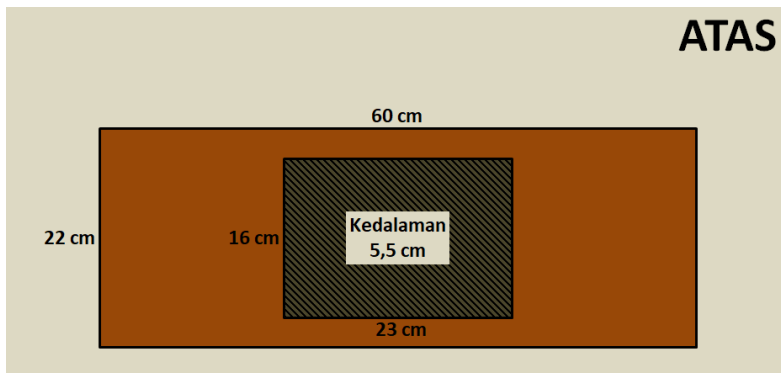
Box alat ini digunakan untuk menyusun komponen dan rangkaian serta digunakan untuk melindungi komponen supaya aman. Pada perancangannya untuk bagian depan box akan dipasang pengaman MCB 3 fasa, kontaktor, *socket input* dan *socket output*. Untuk bagian belakang socket input untuk *power suplay* ac 3 fasa. Sedangkan bagian dalam box akan dipasang, rangkaian power suplay, rangkaian *zero detector*, mikrokontroler ATmega328, rangkaian *solid state relay* dsb. Berdasarkan gambar 3.9 , gambar 3.10, gambar 3.11, gambar 3.12 dan gambar 3.13 berikut



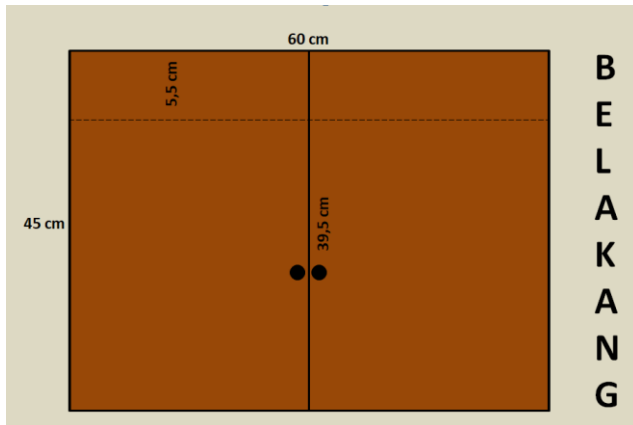
**Gambar 3.9** Perancangan Box Tampak Samping



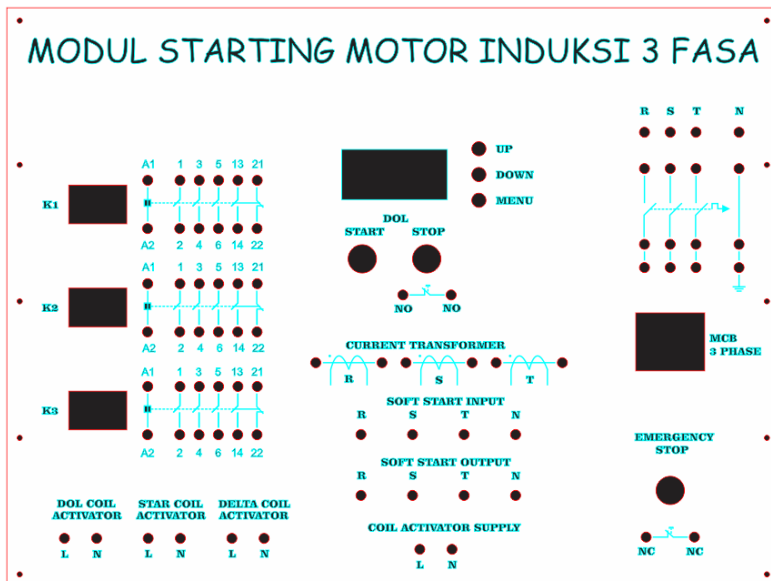
**Gambar 3.10** Perancangan Box Tampak Depan



**Gambar 3.11** Perancangan Box Tampak Atas



**Gambar 3.12** Perancangan Box Tampak Belakang



**Gambar 3.13** Perancangan Desain Akrilik

## **BAB IV**

### **PENGUJIAN DAN ANALISA DATA**

Pada bab ini akan membahas mengenai hasil pengujian dan Analisa atas penyusunan alat untuk “Perancangan *Soft Starting* pada Motor Induksi 3 Fasa Menggunakan Mikrokontroler ATmega328”. Data pengujian sangat diperlukan untuk implementasi dalam dunia nyata. Pengujian system ini untuk mengetahui apakah alat yang sudah dibuat bisa berjalan dengan baik dan dapat diambil analisa dari sistem kerja alat tersebut.

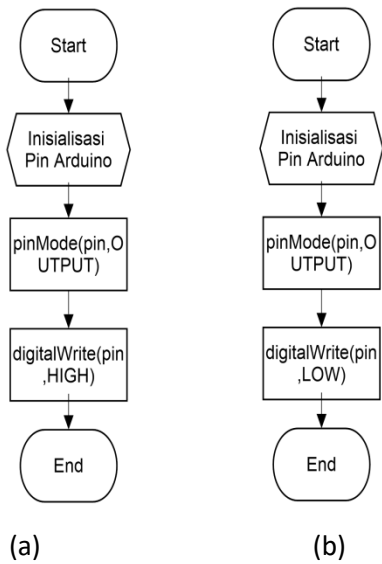
Data pengujian sangat diperlukan untuk kehandalan kerja sistem yang telah dibuat. Pengujian merupakan salah satu langkah yang harus dilakukan untuk mengetahui apakah sistem yang telah dibuat sesuai dengan yang direncanakan. Kesesuaian sistem dengan perencanaan dapat dilihat dari hasil-hasil yang dicapai pada pengujian sistem. Pengujian juga bertujuan untuk mengetahui kelebihan dan kekurangan dari sistem yang telah dibuat. Hasil pengujian tersebut akan dianalisa untuk mengetahui penyebab terjadinya kekurangan atau kesalahan dalam sistem. Pengujian tersebut meliputi :

1. Pengujian modul mikrokontroler Arduino Uno
2. Pengujian *zero cross detector*
3. Pengujian rangkaian PWM
4. Pengujian rangkaian daya
5. Pengujian sistem keseluruhan

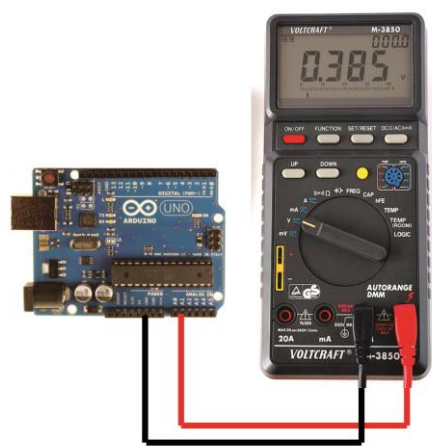
#### **4.1. Pengujian Arduino Uno**

Pengujian ini dilakukan terhadap board Arduino Uno yang diberikan sumber DC 5 volt. Pengujian ini dilakukan agar dapat mengetahui kondisi arduino saat aktif, selain itu dapat diketahui apakah arduino dalam kondisi baik atau tidak. Pengujian dilakukan pada seluruh pin input output yang dimiliki oleh arduino uno yaitu pin A0 s/d A5, D0 s/d D13 milik Arduino Uno dengan cara mengatur keluarannya pada logika 1 dan 0 dengan mengunggah program ke *board* Arduino. Gambar 4.1 (a) merupakan program untuk memberikan tegangan pada pin Arduino berlogika 1 atau sebesar 5 Volt sedangkan pada Gambar 4.1 (b) merupakan program untuk memberikan tegangan keluaran pada pin Arduino berlogika 0 atau 0 Volt. Kemudian memeriksa besar keluaran

tegangan pada setiap pin dan pada pengujian ini seperti pada Gambar 4.2. Data yang didapat adalah seperti pada Tabel 4.1.



Gambar 4.1 Flowchart Pengujian Arduino



Gambar 4.2 Pengujian Arduino



**Tabel 4.1** Hasil Pengujian Arduino Uno

Pin	V Logika 1	V Logika 0
A0	4,99 Volt	0 Volt
A1	4,99 Volt	0 Volt
A2	4,99 Volt	0 Volt
A3	4,99 Volt	0 Volt
A4	4,99 Volt	0 Volt
A5	4,99 Volt	0 Volt
D0	4,99 Volt	0 Volt
D1	4,99 Volt	0 Volt
D2	4,99 Volt	0 Volt
D3	4,99 Volt	0 Volt
D4	4,99 Volt	0 Volt
D5	4,99 Volt	0 Volt
D6	4,99 Volt	0 Volt
D7	4,99 Volt	0 Volt
D8	4,99 Volt	0 Volt
D9	4,99 Volt	0 Volt
D10	4,99 Volt	0 Volt
D11	4,99 Volt	0 Volt
D12	4,99 Volt	0 Volt
D13	4,99 Volt	0 Volt

Berdasarkan data yang didapatkan dari pengujian di atas, nilai nilai tegangan ketika diberi logika 1 mendekati 5V dan ketika diberi logika 0 nilai tegangan 0V. Dengan nilai tersebut, maka dapat disimpulkan bahwa arduino uno ini masih dalam kondisi yang baik dan bisa digunakan pada alat Tugas Akhir.

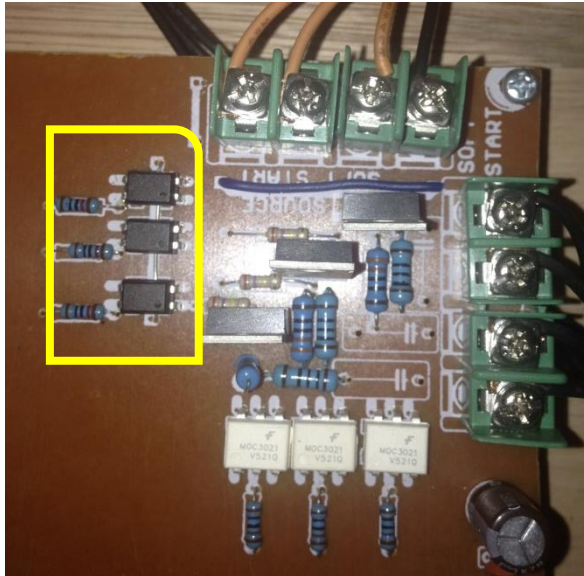
#### 4.2 Pengujian *Zero Cross Detector*

Tujuan pengujian rangkaian adalah untuk mengetahui sinyal gelombang keluaran dari rangkaian *zero cross detector* yang akan digunakan sebagai input pada mikrokontroler sebagai referensi untuk memulai awal proses *Soft Starting*. Sebelum dilakukan pengujian pada rangkaian *zero cross detector*, Dilakukan pengukuran tegangan dengan avometer pada sumber tegangan tiga fasa yang terdapat pada Laboratorium Elektronika Terapan pada Departemen Elektro Automasi. Hasil pengukuran tegangan antar fasa dan fasa netral pada tabel 4.2

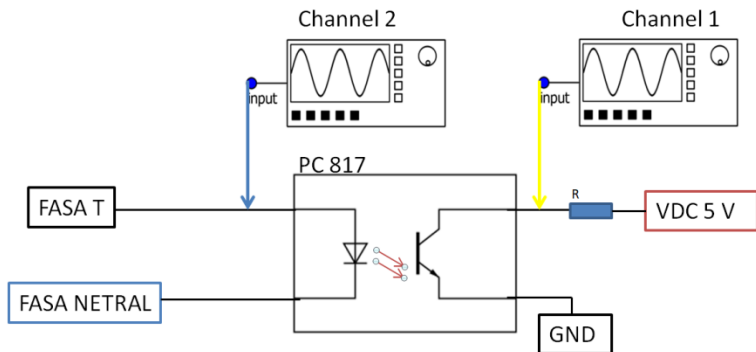
**Tabel 4.2** Hasil Pengukuran Sumber Tiga Fasa *Line to Line*

No	Koneksi antar fasa	Nilai tegangan
1	R-S	403 V
2	S-T	398 V
3	R-T	402 V
4	R-N	225 V
5	S-N	224 V
6	T-N	225 V

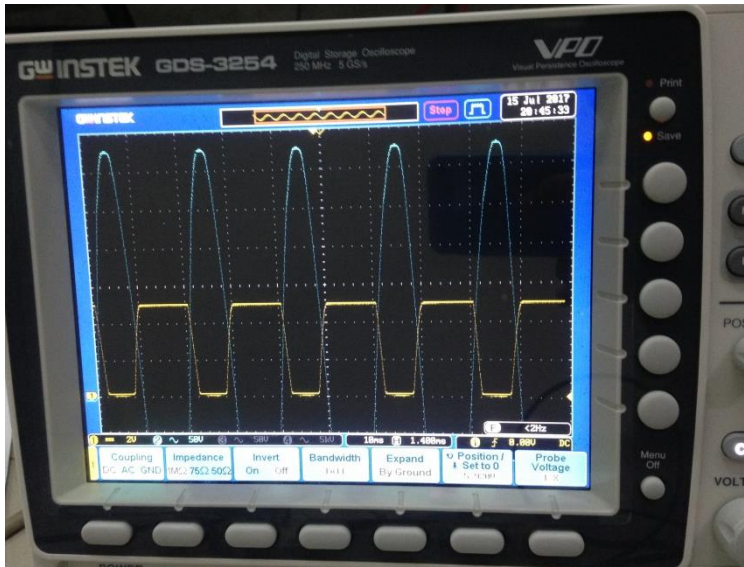
Dapat kita ketahui pada tabel 4.2 bahwa sumber tegangan yang berada pada laboratorium Elektronika Terapan dalam keadaan normal. Kemudian dilakukan pengujian *zero cross detector* dengan menggunakan osiloskop untuk diketahui bentuk gelombangnya. Pada rangkaian ini menggunakan komponen utama yaitu optocoupler tipe PC817. Berikut rangkaian *zero cross detector* pada gambar 4.4 (perhatikan pada tanda kuning) dan gambar 4.5 serta hasil dari rangkaian terdapat pada gambar 4.6



**Gambar 4.4** Rangkaian *Zero Cross Detector* pada Tanda Kuning



**Gambar 4.5** Pengujian *Zero Cross Detector* dengan Osiloskop pada Fasa T



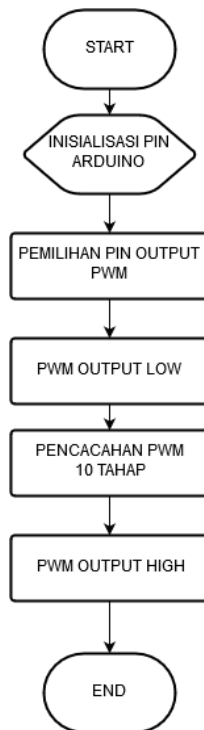
**Gambar 4.6** Sinyal *Input Fasa T-N* dan *Output Zero Cross Detector*

Dapat dilihat pada gambar 4.5 bahwa ketika gelombang positif masuk ke PC817 sehingga led menyala sehingga sensor cahaya transistor akan aktif maka output yang dihasilkan adalah *low*, dan apabila led mati atau keadaan gelombang negatif masuk ke PC817 maka led dan sensor cahaya transistor akan terhalang dan membuat output yang dihasilkan *high*. Sehingga berdasarkan gambar 4.6 bahwa *output* pada channel 1 atau gelombang berwarna kuning *high* ketika gelombang sinus yang berwarna biru negatif. Jadi output *zero cross detector* yang mana gelombang yang berwarna kuning yang nantinya diproses pada Arduino Uno dan bisa digunakan sebagai sinyal input untuk memulai waktu awal penyalan yaitu pada keadaan gelombang berwarna *falling*. Maka bisa disimpulkan bahwa optocoupler tersebut sudah sesuai dengan yang diinginkan sehingga bisa digunakan pada alat Tugas Akhir.

### 4.3 Pengujian Sinyal PWM

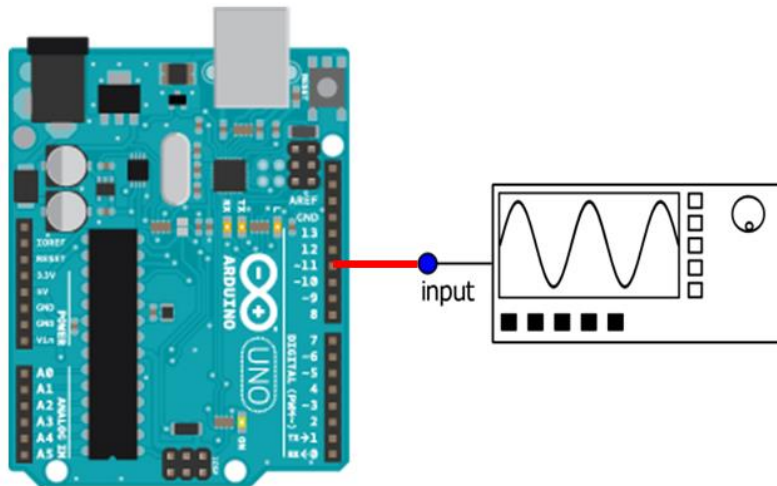
Dapat kita ketahui bahwa PWM (*Pulse Width Modulation*) memanipulasi keluaran digital sedemikian rupa sehingga menghasilkan

sinyal analog. Mikrokontroler mengeset *output* digital ke *LOW* dan *HIGH* bergantian dengan porsi waktu tertentu untuk setiap nilai keluarannya. Durasi waktu untuk nilai *HIGH* disebut *pulse width* atau panjang pulsa. Variasi nilai output analog didapatkan dari perubahan panjang pulsa yang diberikan pada satu periode waktu dan dilakukan berulang-ulang. Tujuan pengujian sinyal PWM pada mikrokontroler ATmega328 ini adalah untuk mengetahui bagaimana hasil dari pemograman yang telah dibuat, dimana program ini akan mendukung *hardware* dari mikrokontroler ATmega328 untuk mengeluarkan sinyal output berupa pulsa *trigger* yang berfungsi untuk penyulutan sudut *alpha* pada triac. Perhatikan gambar 4.7 untuk *flowchart* sinyal PWM.



**Gambar 4.7** Flowchart Sinyal PWM

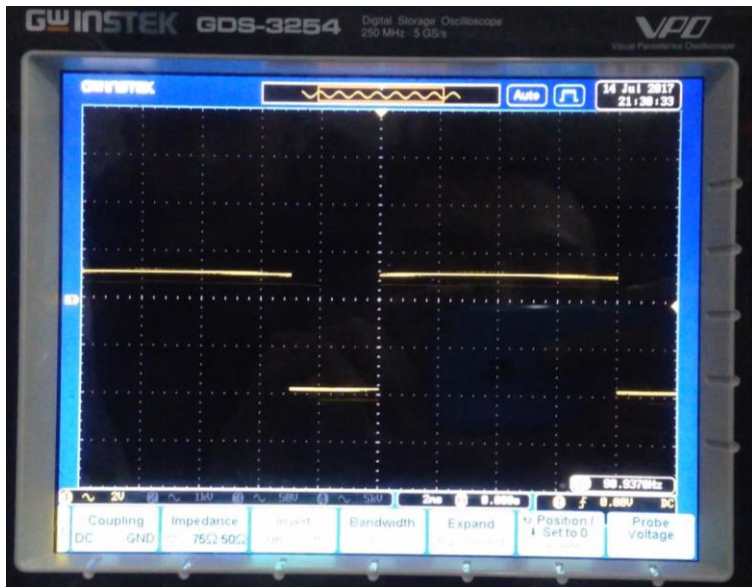
Pada pengujian PWM ini, kami setting dari *LOW* ke *HIGH* dalam 10 tahap. Dengan menggunakan alat ukur osiloskop dan output PWM dari mikrokontroler ATmega328 pada pin 11. Dapat dilihat pada gambar 4.8 untuk pengujian serta hasil gelombang pulsa dari pengujian pada gambar 4.9 dan gambar 4.10.



**Gambar 4.8** Pengujian Sinyal PWM dengan Osiloskop pada Pin 11 Arduino

Adapun pada settingan osiloskop yang kami gunakan adalah sebagai berikut

1. Volt per div adalah 2 volt per div atau 2 volt per kotak
2. Time per div adalah 2 *mili second* per div atau 2 *mili second* per kotak



**Gambar 4.9** Gelombang PWM 30% pada Tahap ke 3

Dapat kita ketahui pada gambar 4.8 bahwa frekuensi yang ditampilkan adalah 98 Hz, sehingga dengan rumus

$$T = 1/f \quad (4.1)$$

dengan :

T = periode / lama waktu dalam satu gelombang

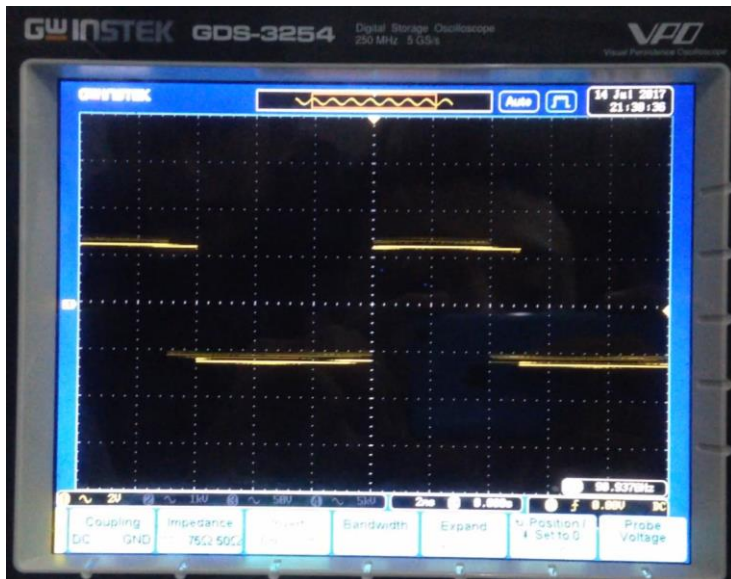
1 = 1 detik

f = frekuensi

jadi hasil lama waktu dalam satu gelombang adalah 10.2 *milisecond* dibulatkan menjadi 10 ms, dapat dilihat kembali pada gambar 4.8 bahwa time/div adalah 2ms yang berarti dalam satu kotak adalah 2ms, sehingga dalam satu gelombang terdapat 5 kotak. Panjang aktif *high* adalah 1.5 kotak maka

$$\frac{1.5}{5} \times 100\% = 30\%$$

Jadi dapat disimpulkan bahwa pada gambar 4.8 gelombang PWM 30% adalah terbukti.



**Gambar 4.10** Gelombang PWM 60% pada Tahap ke 6

Untuk gambar 4.9 panjang aktif high adalah 3 kotak maka

$$\frac{3}{5} \times 100\% = 60\%$$

Jadi dapat disimpulkan bahwa pada gambar 4.8 gelombang PWM 60% adalah terbukti.

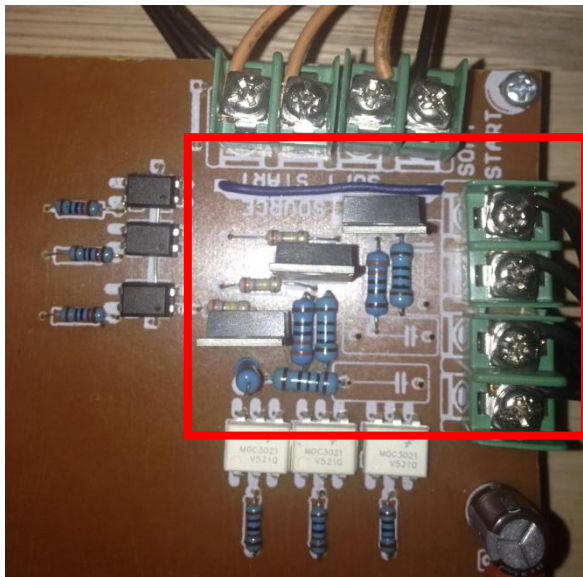
Dari dua sampel pada gambar 4.9 dan 4.10. Maka dapat diambil kesimpulan bahwa sinyal keluaran berupa PWM berjalan dengan baik. Hasil keluaran berupa PWM nantinya yang merupakan proses utama untuk dilakukannya penyulutan triac BT138 atau masuknya tegangan secara bertahap hingga maksimal..

#### 4.4 Pengujian Rangkaian Daya

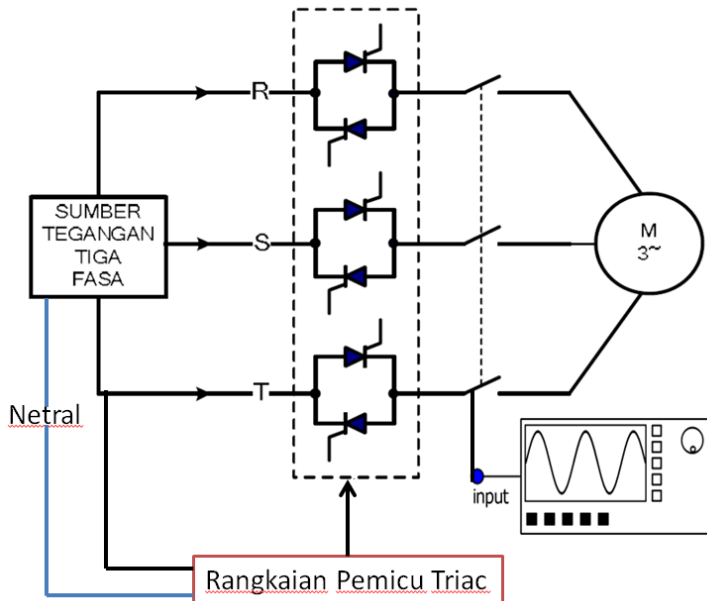
Tujuan pengujian rangkaian daya adalah untuk mengetahui bagaimana fungsi dari komponen setelah terhubung dengan sumber tegangan dan mengetahui bagaimana bentuk gelombang yang terjadi setelah diberikan penyulutan terhadap TRIAC BT138. sebagai pembatas



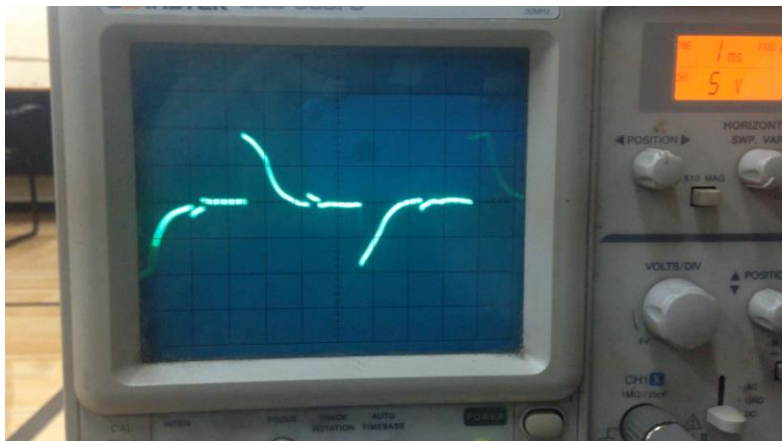
antara rangkaian daya dengan rangkaian kontrol terdapat rangkaian pemisah dengan komponen utama yaitu optocoupler MOC 3021 yang berguna sebagai pelindung antara rangkaian kontrol dengan rangkaian daya. Untuk media pengukuran yang digunakan pada pengujian rangkaian daya ini adalah avometer dan osiloskop, guna sebagai mengetahui penyulutan gelombang sinus dalam rentang waktu tertentu. Adapun pengujian rangkaian daya ini dilakukan pada salah satu fasa dengan netral yaitu fasa T-N dengan rentang waktu 15 detik. Dapat dilihat pada rangkaian gambar 4.12 dan gambar 4.13 untuk titik pengujian serta hasil gelombang penyulutan triac dari pengujian pada gambar 4.14 , gambar 4.15 , gambar 4.16 dan gambar 4.17



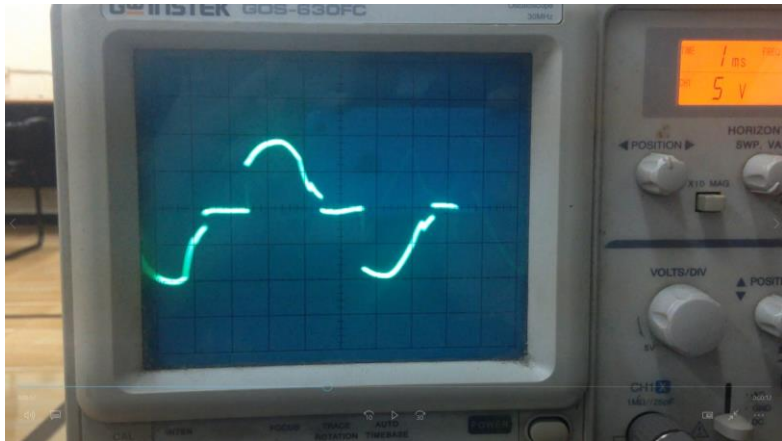
**Gambar 4.12** Rangkaian Daya dengan Tanda Warna Merah



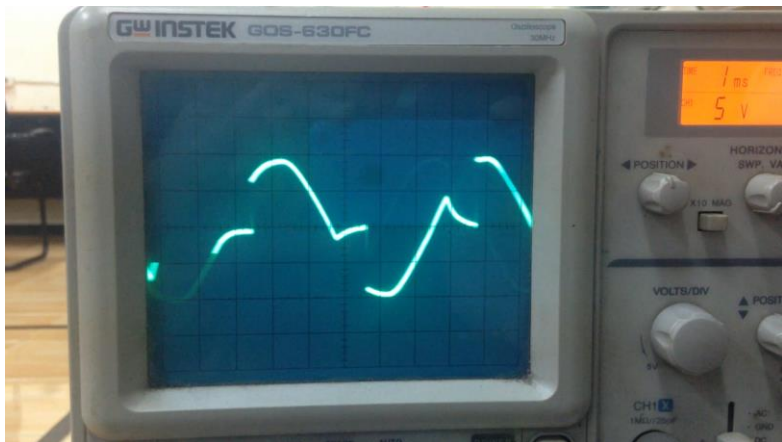
**Gambar 4.13** Pengujian Rangkaian Daya dengan Osiloskop



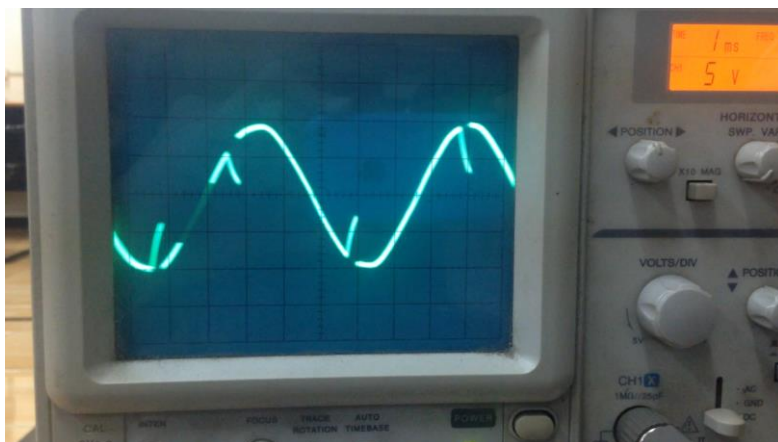
**Gambar 4.14** Gelombang Penyulutan *Triac* Bertegangan 55 Volt



**Gambar 4.15** Gelombang Penyulutan *Triac* Bertegangan 175 Volt



**Gambar 4.16** Gelombang Penyulutan *Triac* Bertegangan 206 Volt



**Gambar 4.17** Gelombang Penyulutan *Triac* Bertegangan 225 Volt

Dapat kita ketahui dari hasil keluaran sudut penyulutan triac pada gambar 4.13, gambar 4.14, gambar 4.15 dan gambar 4.16, bahwa dalam hasilnya kurang sesuai yang diharapkan dikarenakan masih terlihat kasar dalam pembentukan gelombang sempurna sinus tegangan fasaneutral. Selain dari pengamatan dengan melihat gambar gelombang tegangan yang dihasilkan, kami juga melakukan pengukuran keluaran tegangan hasil penyulutan triac. Untuk waktu penyulutan triac kami set 15 detik. Berikut tabel 4.3 hasil pengukuran dari masing masing fasa ke netral waktu penyulutan triac tanpa beban.

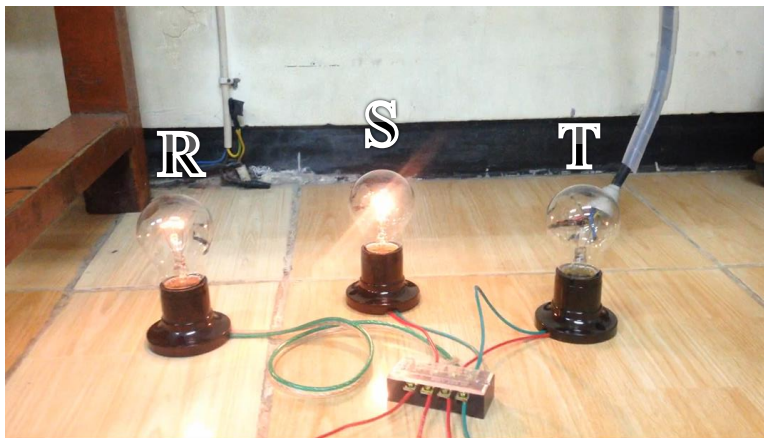
**Tabel 4.3** Hasil Pengukuran Waktu Penyulutan *Triac* Tanpa Beban

Waktu Penyalan(s)	Fasa R-N (V)	Fasa S-N (V)	Fasa T-N (V)
0.1 s	41 V	75 V	55 V
3 s	101 V	117 V	128 V
5 s	161 V	160 V	159 V
8 s	186 V	195 V	189 V
10 s	211 V	212 V	206 V
13 s	227 V	222 V	221 V
15 s	229 V	226 V	225 V

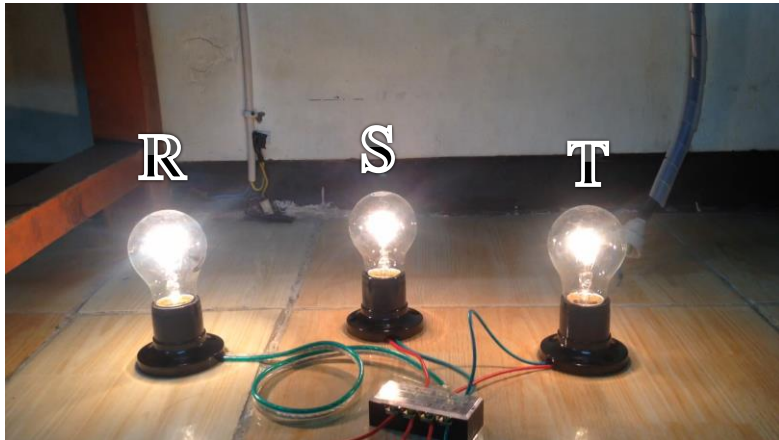
Pengujian *soft starter* dengan menggunakan triac BT138 dengan rentang waktu 15 detik dilakukan dengan menggunakan beban tiga buah lampu pijar yang dihubungkan dengan triac dari fasa R, fasa S dan fasa T dengan masing masing lampu dihubungkan ke netral tiga fasa. Yang hasilnya lampu pijar tersebut mulai nyala pada kondisi redup dan tahap demi tahap lampu dapat nyala pada kondisi normalnya. Dapat dilihat pada tabel 4.4 untuk hasil pengukuran dan pada gambar 4.18 , gambar 4.19 dan gambar 4.20 untuk hasil penyalaaan lampu.

**Tabel 4.4** Hasil Pengukuran Waktu Penyulutan *Triac* dengan Lampu

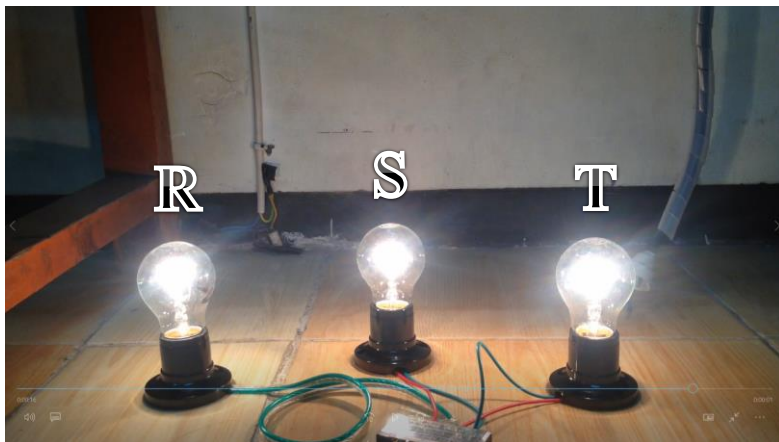
Waktu Penyalan(s)	Fasa R-N (V)	Keadaan Lampu	Fasa S-N (V)	Keadaan Lampu	Fasa T-N (V)	Keadaan Lampu
0.1 s	101 V	redup	111 V	redup	105V	belum nyala
3 s	130 V	redup	139 V	redup	141 V	redup
8 s	205 V	agak terang	208 V	agak terang	197 V	agak terang
15 s	222 V	terang	223 V	terang	222 V	terang



**Gambar 4.18** Penyalaaan Lampu Secara *Soft Start* pada 0.1 Detik



**Gambar 4.19** Penyalaan Lampu Secara *Soft Start* pada 8 Detik



**Gambar 4.20** Penyalaan Lampu Secara *Soft Start* pada 15 Detik

Dan setelah dilakukan pengujian triac dapat berfungsi dan dapat melakukan proses penyulutan sinus dengan baik yang dapat kita ketahui pada gambar 4.17, gambar 4.18 dan gambar 4.19. Selanjutnya dapat dilihat pada tabel 4.5 untuk hasil pengukuran penyulutan *triac* secara *line to line*.

**Tabel 4.5** Hasil Pengukuran Penyulutan *Triac* Line R dengan Line T

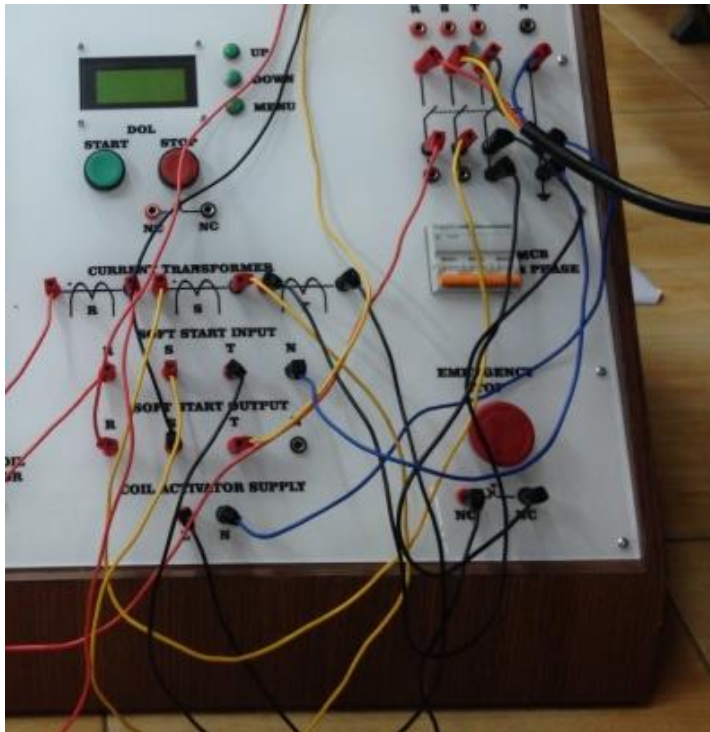
No	Waktu Penyalaan(s)	Tegangan (V)
1	1 s	5 V
2	3 s	6 V
3	4 s	116 V
4	5 s	151 V
5	8 s	249 V
6	10 s	326 V
7	13 s	396 V
8	15 s	403 V

Dapat kita ketahui hasil dari tabel 4.4 untuk penyulutan *triac* pada detik ke-1 hingga detik ke-3 tegangan masuk kecil sehingga kemungkinan nanti akan mengganggu pada saat *Soft Starting* dengan beban motor induksi tiga fasa.

#### 4.5 Pengujian Sistem Keseluruhan

Setelah melalui beberapa pengujian pada masing-masing bagian dari Alat Tugas akhir ini, maka terakhir adalah melakukan pengujian keseluruhan alat. Pada pengujian keseluruhan ini akan diuji mengenai sistem kerja secara keseluruhan. Sekaligus nantinya untuk mengetahui bagaimana karakteristik arus motor yang diasut dengan alat yang telah dibuat. Lihat gambar 4.21 dan gambar 4.22 untuk *wiring* secara *soft start* pada modul *starting* motor induksi tiga fasa

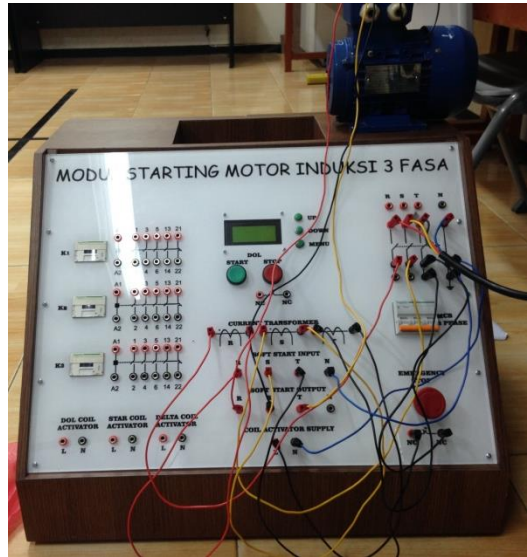




**Gambar 4.21** *Wiring Soft Starting Motor Induksi Tiga Fasa*

Dapat dilihat dari gambar 4.21 bahwa wiring dari sumber tiga fasa masuk melewati sensor arus sebagaimana yang telah dibuat oleh rekan saya. Kemudian masuk ke *SOFT START INPUT* yang didalamnya akan bercabang dua masukan, yang satu menuju salah satu kaki triac BT138, dan masukan yang lain melewati rangkaian *zero crossing detector* dan rangkaian kontrol hingga menuju kaki *gate* triac BT138. Lalu dilakukan wiring dari *SOFT START OUTPUT* menuju ke beban motor induksi tiga fasa. Untuk *power supply* Arduino Uno dari fasa T dan Netral yang didalamnya terdapat trafo *step down* dan rangkaian *power supply*.





**Gambar 4.22** *Wiring Soft Starting Motor Induksi Tiga Fasa*

Sistem kerja alat kami ini dimulai saat suplai dari sumber tiga fasa PLN menyala dan sumber fasa T dan netral untuk *power supply* pada modul kontrol Arduino Uno. Saat kondisi tersebut sudah dipenuhi, LCD pun menyala dan menunjukkan tampilan seperti pada Gambar 4.23.



**Gambar 4.23** *Menu Home pada LCD*

Pada Gambar 4.21, terlihat ada tiga jenis tombol “UP”, “DOWN” dan “MENU” yang memiliki fungsinya masing masing dan pada LCD menunjukkan tampilan *home*. Untuk proses selanjutnya apabila tombol “MENU” ditekan sekali maka tampilan LCD akan berubah dan akan muncul seperti pada gambar 4.24



**Gambar 4.24** Empat Menu Display pada LCD

Terlihat pada gambar 4.24, terdapat tampilan dengan 4 *menu*. Yang mana masing masing *menu* tersebut memiliki kegunaan masing masing dan hanya *menu* “Set Soft Start” dan *menu* “Soft Starting” yang memiliki keterkaitan. Mengawali dengan *menu* “Set Soft Start” dengan menggerakkan arah panah dengan tombol “DOWN” kemudian tekan “MENU” yang mana tombol tersebut juga dapat digunakan sebagai tombol ok. Lalu LCD akan muncul dengan tampilan seperti gambar 4.25



**Gambar 4.25** Tampilan Menu “Set Soft Start”

Kemudian dapat diatur rentang waktu untuk dilakukan *Soft Starting* sesuai keinginan, akan tetapi dengan batas angka 5.0 detik hingga 25.0 detik. Lalu tekan tombol “MENU” lagi dan muncul tampilan pada gambar 4.26



**Gambar 4.26** Tampilan Menu “Set Soft Start” Setelah Set Time

Secara otomatis setelah ditekan tombol “MENU” akan kembali ke tampilan seperti pada gambar 4.23, kemudian dapat dipilih pada menu “Soft Starting”, sehingga akan muncul seperti gambar 4.27 berikut.



**Gambar 4.27** Tampilan Menu “Soft Starting”

Kemudian untuk memulai *Soft Starting* dapat menekan tombol menu sehingga proses *Soft Starting* berjalan dari 0% hingga 100% , yang mana ketika ingin menghentikan *Soft Starting* yang telah dilakukan penyulutan hingga 100% dapat menekan tombol “MENU”. Untuk tampilan proses *Soft Starting* dapat diamati pada gambar 4.28

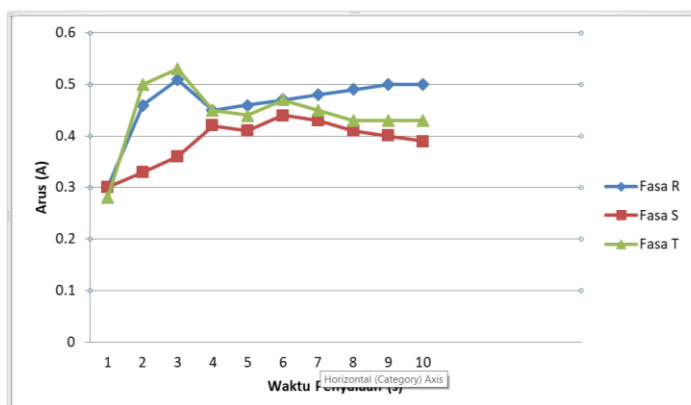


**Gambar 4.28** Tampilan Menu “Soft Starting” Penyulutan Maksimal

Selanjutnya adalah mengetahui karakteristik arus motor dengan menggunakan metode *Soft Starting*, berikut kami berikan 4 jenis rentang waktu penyulutan *triac* yaitu 10 detik, 15 detik, 20 detik dan 25 detik. Dengan beban motor induksi tiga fasa 1/4 HP dengan konfigurasi delta tanpa dikopel dengan beban lain. Berikut hasilnya terdapat pada tabel 4.6, tabel 4.7, tabel 4.8 dan tabel 4.9. serta gambar grafik dari masing – masing tabel pada gambar 4.29, gambar 4.30, gambar 4.31 dan gambar 4.32.

**Tabel 4.6** *Soft Starting* Dengan Rentang Waktu 10 Detik

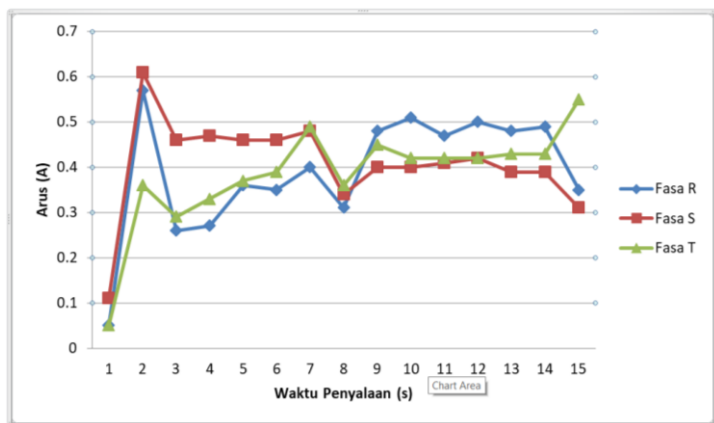
Waktu Penyalan (s)	Arus R (A)	Arus S (A)	Arus T (A)
1 s	0.30	0.30	0.28
2 s	0.46	0.33	0.50
3 s	0.51	0.36	0.53
4 s	0.45	0.42	0.45
5 s	0.46	0.41	0.44
6 s	0.47	0.44	0.47
7 s	0.48	0.43	0.45
8 s	0.49	0.41	0.43
9 s	0.50	0.40	0.43
10 s	0.50	0.39	0.43



**Gambar 4.29** Grafik Arus *Soft Starting* 10 Detik

**Tabel 4.7** *Soft Starting* Dengan Rentang Waktu 15 Detik

Waktu Penyalan (s)	Arus R (A)	Arus S (A)	Arus T (A)
1 s	0.05	0.11	0.05
2 s	0.57	0.61	0.36
3 s	0.26	0.46	0.29
4 s	0.27	0.47	0.33
5 s	0.36	0.46	0.37
6 s	0.35	0.46	0.39
7 s	0.40	0.48	0.49
8 s	0.31	0.34	0.36
9 s	0.48	0.40	0.45
10 s	0.51	0.40	0.42
11 s	0.47	0.41	0.42
12 s	0.50	0.42	0.42
13 s	0.48	0.39	0.43
14 s	0.49	0.39	0.43
15 s	0.35	0.31	0.55

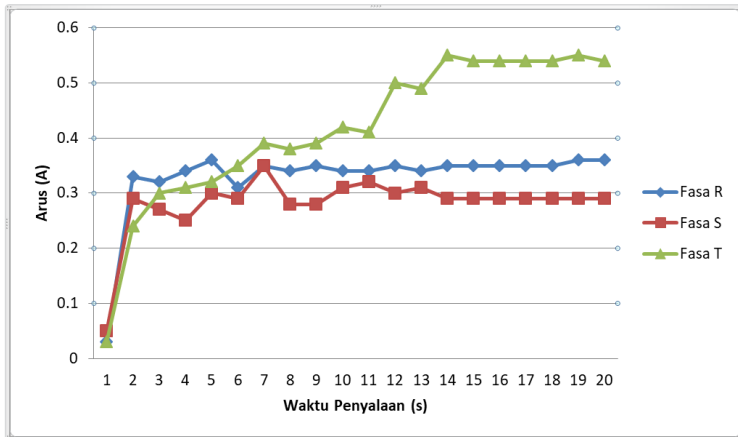


**Gambar 4.30** Grafik Arus *Soft Starting* 15 Detik

**Tabel 4.8** *Soft Starting* Dengan Rentang Waktu 20 Detik

Waktu Penyalaan (s)	Arus R (A)	Arus S (A)	Arus T (A)
1 s	0.03	0.05	0.03
2 s	0.33	0.29	0.24
3 s	0.32	0.27	0.30
4 s	0.34	0.25	0.31
5 s	0.36	0.30	0.32
6 s	0.31	0.29	0.35
7 s	0.35	0.35	0.39
8 s	0.34	0.28	0.38
9 s	0.35	0.28	0.39
10 s	0.34	0.31	0.42
11 s	0.34	0.32	0.41
12 s	0.35	0.30	0.50
13 s	0.34	0.31	0.49
14 s	0.35	0.29	0.55
15 s	0.35	0.29	0.54
16 s	0.35	0.29	0.54

17 s	0.35	0.29	0.54
18 s	0.35	0.29	0.54
19 s	0.36	0.29	0.55
20 s	0.36	0.29	0.54

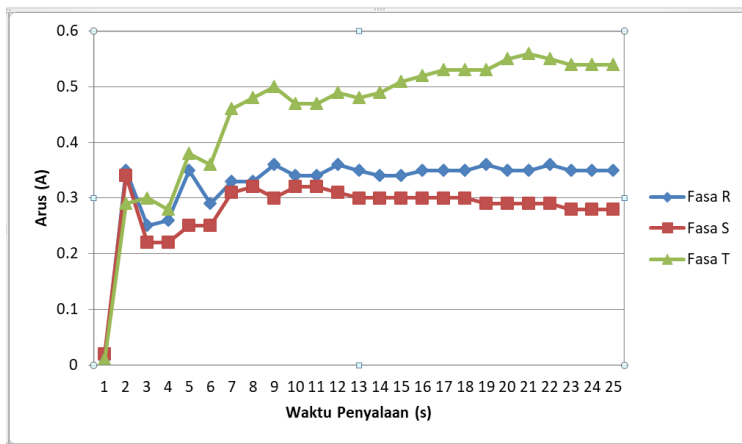


**Gambar 4.31** Grafik Arus *Soft Starting* 20 Detik

**Tabel 4.9** *Soft Starting* Dengan Rentang Waktu 25 Detik

Waktu Penyalaaan (s)	Arus R (A)	Arus S (A)	Arus T (A)
1 s	0.01	0.02	0.01
2 s	0.35	0.34	0.29
3 s	0.25	0.22	0.30
4 s	0.26	0.22	0.28
5 s	0.35	0.25	0.38
6 s	0.29	0.25	0.36
7 s	0.33	0.31	0.46
8 s	0.33	0.32	0.48
9 s	0.36	0.30	0.50
10 s	0.34	0.32	0.47
11 s	0.34	0.32	0.47

12 s	0.36	0.31	0.49
13 s	0.35	0.30	0.48
14 s	0.34	0.30	0.49
15 s	0.34	0.30	0.51
16 s	0.35	0.30	0.52
17 s	0.35	0.30	0.53
18 s	0.35	0.30	0.53
19 s	0.36	0.29	0.53
20 s	0.35	0.29	0.55
21 s	0.35	0.29	0.56
22 s	0.36	0.29	0.55
23 s	0.35	0.28	0.54
24 s	0.35	0.28	0.54
25 s	0.35	0.28	0.54



**Gambar 4.32** Grafik Arus *Soft Starting* 25 Detik

Setelah dilakukan pengujian alat dengan menggunakan motor, didapat hasil bahwa proses pencacahan tegangan secara bertahap untuk sistem kerja *Soft Starting* telah terpenuhi. Tetapi untuk karakteristik arus yang diinginkan untuk menjadi lebih halus belum terpenuhi.



Dari grafik gambar 4.29 dapat dianalisa bahwa pemberian *duty cycle* 10% hingga 100% berdampak pada tegangan yang mulanya kecil menjadi besar atau maksimal. Maka tegangan berbanding lurus dengan *duty cycle*. Sebaliknya berbanding terbalik terhadap pemberian *delay* yang semakin rendah maka tegangan semakin tinggi. Dikarenakan rentang waktu penyulutan waktu *delay* yang semakin kecil maka akan memberikan gelombang sinusoidal yang penuh. dapat dianalisa arusnya bahwa pemberian *duty cycle* 10% hingga 100% berdampak pada arus yang mulanya diberi *duty cycle* 10% Arus yang terukur 0,30A pada fasa S akan naik pada puncak yang terukur 0.44 A ketika diberi *duty cycle* 60% disebabkan karena arus *starting* yang berulang-ulang, Tetapi akan turun pada 0,3 A ketika diberi *duty cycle* 100 % disebabkan karena pemucuan triac yang terdelay 0 mS dari alat. Bisa dibandingkan dengan grafik gambar 4.31 dan gambar 4.32, semakin lama *delay* antar tahap yang diberikan maka akan lebih baik tahap tahap dilakukan pencacahan sinus.

Akan tetapi bisa kita ketahui bahwa motor induksi tiga fasa memiliki tegangan kerja nominal untuk memulai mengaktifkan motor. Jadi apabila rentang waktu *Soft Starting* lebih lama, maka tegangan pada angka 0 – 150 V akan lebih lama, sehingga pada angka voltase tersebut motor tidak dapat melakukan start dengan baik. Dapat diketahui pada gambar 4.32 pada detik ke 2, arus langsung melebihi arus nominal, itu dikarenakan tegangan yang masuk di optimal sehingga menyebabkan *starting* motor menjadi buruk.

*Halaman ini sengaja dikosongkan*

## **BAB V**

### **PENUTUP**

Bab penutup ini berisi kesimpulan yang diperoleh selama proses pembuatan tugas akhir yang berjudul “Perancangan *Soft Starting* pada Motor Induksi 3 Fasa Menggunakan Mikrokontroler ATmega328” serta saran untuk pengembangan alat ini kedepannya.

#### **5.1 Kesimpulan**

Berdasarkan hasil pengujian pada Tugas Akhir ini, maka diperoleh beberapa kesimpulan yaitu

1. Modul *Soft Starting* yang dibuat bisa digunakan untuk motor induksi 3 fasa dengan spesifikasi maksimal 2 HP untuk menjaga komponen dari arus start yang dihasilkan, dengan lama waktu proses *Soft Starting* maksimal 25 detik dan memiliki rating kerja alat dengan arus maksimal 12 A dan tegangan maksimal 600 V AC, rating ini berdasarkan rating *triac* sebagai komponen utama rangkaian daya kendali tegangan AC.
2. *Soft starter* adalah metode *starting* lain untuk motor induksi 3 fasa yang memiliki keandalan yang lebih baik untuk mengatasi arus lebih pada saat start motor. Akan tetapi motor tiga fasa yang digunakan terlalu kecil daya nya, sehingga arus *starting* yang dihasilkan tidak terlalu menonjol.
3. Sebuah mikrokontroler ATmega328 berplatform arduino sebagai kontrol utama alat dan dapat mengatur dengan baik proses pemicuan ketiga buah *triac* sehingga dapat berjalan serempak.
4. *Duty cycle* dimulai dari 0% membuat penyulutan *triac* dimulai 0 derajat sehingga tegangan yang masuk dari 0 V sangat tidak disarankan karena motor induksi 3 fasa memiliki tegangan kerja tersendiri untuk motor dapat start dengan baik.

#### **5.2 Saran**

Beberapa saran yang kami usulkan untuk mengembangkan tugas akhir kami adalah

1. Pemilihan motor induksi tiga fasa bisa minimal berspesifikasi 1 HP atau 750 kw, sehingga hasil dari arus dapat lebih bisa dianalisa dan dibandingkan dengan metode DOL atau Wye-

Delta. Dan juga bisa motor induksi tiga fasa bisa dikopel dengan motor lain atau generator, guna menambah variasi analisa arus.

2. Ketika dilakukan *Soft Starting* dengan beban motor, setelah penyulutan triac maksimal, sangat disarankan untuk berpindah ke kontaktor, supaya triac BT138 tidak *overheat* apabila penyalan beban motor dalam waktu yang lama.
3. Sangat disarankan ketika duty cycle keluaran mikrokontroler Atmega328 dimulai dari 25%, karena tegangan yang dihasilkan sudah mencapai rata rata 180 V.

Semoga saran saran yang kami sebutkan dapat bermanfaat dikemudian hari, dan semoga alat yang kami buat ini dapat berguna untuk jurusan sebagai media pembelajaran atau media praktikum untuk mata kuliah mesin mesin listrik.

## DAFTAR PUSTAKA

- [1] Syahwil, Muhammad. 2013. “Panduan Mudah Simulasi & Praktek Mikrokontroler Arduino”. Yogyakarta: ANDI
- [2] Wikipedia, 2017, “Motor Listrik” [https://en.wikipedia.org/wiki/motor\\_listrik](https://en.wikipedia.org/wiki/motor_listrik) (diakses tanggal 28 Mei 2017)
- [3] Datasheet MOC3021, 1995, “6-Pin DIP Random Phase Optoisolator Triac Driver Output”, Motorola <http://www.alldatasheet.com/> (23 Juni 2017)
- [4] Datasheet PC817, 2000, “PC817 Series”, Sharp, <http://www.alldatasheet.com/> (20 Juni 2017)
- [5] Datasheet BT138, 2001, “BT138 Series”, NXP, <http://www.alldatasheet.com/> (15 Juni 2017)
- [6] Basuki Sugiharto , Agung , *Soft Starting dan Dynamic Braking Pada Motor Induksi Tiga Fasa Menggunakan Mikrokontroler AT89S51*, Skripsi S-1, Teknik Elektro, Universitas Diponegoro, Semarang, 2005 .
- [7] Dwi Riyadi, *Soft Starting Pada Motor Induksi 3 Fasa*, Teknik Elektro, Universitas Diponegoro, Semarang, 2001.
- [8] Lister, *Rangkaian dan Mesin Listrik*, Penerbit Erlangga, Jakarta, 1993.
- [9] Agus Sutriyono, Rancang Bangun Pengendali Motor 1 Fasa Dengan Metode *Zero Crossing Detector* Berbasis Arduino, Skripsi S-1, Teknik Elektro, Universitas Muhammadiyah Surakarta, Surakarta, 2017

*Halaman ini sengaja dikosongkan*

## LAMPIRAN A

### A.1 Listing Program Pada Arduino

```
//#include <EmonLib.h>
//EnergyMonitor arus;

#include <LiquidCrystal.h>
LiquidCrystal lcd (A5, A4, A3, 13, 12, 2);

////definisi ADC

// Voltage Reference: AVCC pin
#define ADC_VREF_TYPE ((0<<REFS1) | (1<<REFS0) | (0<<ADLAR))

// Voltage Reference: AREF pin
#define ADC_VREF_TYPE ((0<<REFS1) | (0<<REFS0) | (0<<ADLAR))

// Voltage Reference: Int., cap. on AREF
#define ADC_VREF_TYPE ((1<<REFS1) | (1<<REFS0) | (0<<ADLAR))

// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
    ADMUX = adc_input | ADC_VREF_TYPE;
    // Delay needed for the stabilization of the ADC input voltage
    //delay_us(10);
    //delayMicroseconds(10);
    // Start the AD conversion
    ADCSRA |= (1 << ADSC);
    // Wait for the AD conversion to complete
    while ((ADCSRA & (1 << ADIF)) == 0);
    ADCSRA |= (1 << ADIF);
    return ADCW;
}

void adcInit(void) {
```

```

// ADC initialization
// ADC Clock frequency: 1000,000 kHz
// ADC Voltage Reference: AVCC pin
// ADC Auto Trigger Source: ADC Stopped
// Digital input buffers on ADC0: On, ADC1: On, ADC2: On, ADC3:
On
// ADC4: On, ADC5: On, ADC6: On, ADC7: On

//DIDR0 = (0 << ADC7D) | (0 << ADC6D) | (0 << ADC5D) | (0 <<
ADC4D) | (0 << ADC3D) | (0 << ADC2D) | (0 << ADC1D) | (0 <<
ADC0D);

DIDR0 = (0 << ADC5D) | (0 << ADC4D) | (0 << ADC3D) | (0 <<
ADC2D) | (0 << ADC1D) | (0 << ADC0D);
ADMUX = ADC_VREF_TYPE;
ADCSRA = (1 << ADEN) | (0 << ADSC) | (0 << ADATE) | (0 <<
ADIF) | (0 << ADIE) | (1 << ADPS2) | (0 << ADPS1) | (0 << ADPS0);
ADCSRB = (0 << ADTS2) | (0 << ADTS1) | (0 << ADTS0);

}

////

//output pin
const int StartSTARpin = 5;
const int StartDELTApin = 4;

const int StartRSSpin = 11;
const int StartSSSpin = 10;
const int StartTSSpin = 9;

//input pin
const int UPpin = 3;
const int DOWNpin = 1;
const int MENUpin = 0;

//analog pin
/*
const int CurrentRpin = A0;

```



```

const int CurrentSpin = A1;
const int CurrentTpin = A2;
*/
#define currentRpin 0
#define currentSpin 1
#define currentTpin 2

const int ZCRpin = 6;
const int ZCSpin = 7;
const int ZCTpin = 8;

String txtDOL    = "Direct Online ";
String txtStartDelta = "Star Delta ";
String txtSoftStart = "Soft Starting ";
String txtSet     = "Set Soft Start";

#include <FlexiTimer2.h>

int dutySet = 10;

int cDutyT = 0;
int dutyT = dutySet;
boolean gasspollT = false;

int cDutyS = 0;
int dutyS = dutySet;
boolean gasspollS = false;

int cDutyR = 0;
int dutyR = dutySet;
boolean gasspollR = false;

void onDuty(void) {
  if (gasspollT == true) {
    cDutyT++;
    if (cDutyT > dutyT) {
      digitalWrite(StartTSSpin, LOW);
    }
  }
}

```

```

    else {
        digitalWrite(StartTSSpin, HIGH);
    }
    if (cDutyT > 10)cDutyT = 0;
}

if (gasspollS == true) {
    cDutyS++;
    if (cDutyS > dutyS) {
        digitalWrite(StartSSSpin, LOW);
    }
    else {
        digitalWrite(StartSSSpin, HIGH);
    }
    if (cDutyS > 10)cDutyS = 0;
}

if (gasspollR == true) {
    cDutyR++;
    if (cDutyR > dutyR) {
        digitalWrite(StartRSSpin, LOW);
    }
    else {
        digitalWrite(StartRSSpin, HIGH);
    }
    if (cDutyR > 10)cDutyR = 0;
}
}

void setup()
{
    //Setting I/O
    pinMode(UPpin, INPUT_PULLUP);
    pinMode(DOWNpin, INPUT_PULLUP);
    pinMode(MENUpin, INPUT_PULLUP);

    pinMode(StartSTARpin, OUTPUT);
    pinMode(StartDELTApin, OUTPUT);
    pinMode(StartRSSpin, OUTPUT);
}

```

```

pinMode(StartSSSpin, OUTPUT);
pinMode(StartTSSpin, OUTPUT);

digitalWrite(StartSTARpin, HIGH);
digitalWrite(StartDELTApin, HIGH);
digitalWrite(StartRSSpin, HIGH);
digitalWrite(StartSSSpin, HIGH);
digitalWrite(StartTSSpin, HIGH);

Serial.begin(115200);
Serial.println("Programmable Motor Starting");
Serial.println("Activated !!");
Serial.end();

//Timer1.initialize(50000);
//FlexiTimer2::set(0.5, 1.0 / 2000, onDuty); // call every 500 1ms
"ticks"
FlexiTimer2::set(1, onDuty); // MsTimer2 style is also supported

//FlexiTimer2::start();

/* add setup code here */
adcInit();

//arus.current(CurrentRpin, 111.1);

lcd.begin(16, 4);
lcd.clear();
/*
lcd.setCursor(0, 0);
lcd.print("Ya..Meitahh..");
lcd.setCursor(0, 1);
lcd.print("test");
lcd.setCursor(0, 2);
lcd.print("1");
lcd.setCursor(0, 3);
lcd.print("oke");
*/
}

```

```

int cTombol = 0;
long tombolMillis = 0;
long dMillis = 0;
signed char baris = 0;
boolean sBlink = false;

///punyae soft
long softMillisR = 0;
long softMillisS = 0;
long softMillisT = 0;

float lamaStart = 25.0;

boolean softEnable = false;

boolean atHome = true;

/////untuk debounce
int dMax=5000, dLimit=4500, dValue=4000;

/////Arus
float arusR = 0.0, arusS = 0.0, arusT = 0.0;
float arusMaxR = 0.0, arusMaxS = 0.0, arusMaxT = 0.0;
int maxR = 0, maxS = 0, maxT = 0;
int minR = 1023, minS = 1023, minT = 1023;

void loop()
{
  if (millis() - dMillis > 1000) {
    dMillis = millis();
    if (atHome) {
      displayHome();
    }
    else {
      tampilan();
    }
  }
}

```

```

if (atHome == true) {
  if (digitalRead(MENUpin) == LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {
      while (digitalRead(MENUpin) == LOW) {
        lcd.clear();
      }
      atHome = false;
      tampilan();
    }
  }
  else {
    cTombol = 0;
  }
}
else {
  readKey();
}

}

void displayValue(void) {
  if (millis() - dMillis > 500) {
    dMillis = millis();

    lcd.setCursor(0, 1);
    lcd.print("R:");
    lcd.print(arusR);
    lcd.print("A ");
    lcd.print(arusMaxR);
    lcd.print("A ");

    lcd.setCursor(0, 2);
    lcd.print("S:");
    lcd.print(arusS);
    lcd.print("A ");
    lcd.print(arusMaxS);
  }
}

```

```

    lcd.print("A ");

    lcd.setCursor(0, 3);
    lcd.print("T:");
    lcd.print(arusT);
    lcd.print("A ");
    lcd.print(arusMaxT);
    lcd.print("A ");
}
}

void modeDOL(void) {

    while (digitalRead(MENUUpin) == LOW) {
        lcd.clear();
    }
    lcd.setCursor(0, 0);
    lcd.print(txtDOL);
    while (1) {
        readArus();
        displayValue();
        //double Irms = arus.calcIrms(1480);
        //lcd.setCursor(0, 1);
        //lcd.print(" Arus R:");
        //lcd.print(Irms * 230.0);

        if (digitalRead(UPpin) == LOW && digitalRead(DOWNpin) ==
LOW) {
            cTombol++;
            if (cTombol >= dMax)cTombol = dLimit;
            if (cTombol == dValue) {
                //atHome = true;
                lcd.clear();
                tampilan();
                arusMaxR = 0.0;
                arusMaxS = 0.0;
                arusMaxT = 0.0;
            }
        }
    }
}

```

```

        while (digitalRead(UPpin) == LOW && digitalRead(DOWNpin)
== LOW);
        break;
    }
}
else {
    cTombol = 0;
}
}
}
}

```

```

boolean statusStarDelta = false;
uint8_t sdc = 0;
long starDeltaMillis = 0;
long cDownMillis = 0;
long timeStarDelta = 5000;//order mili detik
void modeStarDelta(void) {
    while (digitalRead(MENUpin) == LOW) {
        lcd.clear();
    }
    lcd.setCursor(0, 0);
    lcd.print(txtStartDelta);
    lcd.setCursor(13, 0);
    lcd.print('[');
    lcd.print(timeStarDelta / 1000);
    lcd.print(']');
}

```

```

while (1) {

    if (digitalRead(MENUpin) == LOW) {
        cTombol++;
        if (cTombol >= dMax)cTombol = dLimit;
        if (cTombol == dValue) {
            lcd.begin(16, 4);
            //lcd.clear();
            //lcd.setCursor(0, 0);
            //lcd.print("
");

```

```

    if (statusStarDelta == false) {
        sdc = 0;
        //lcd.begin(16, 4);
        lcd.setCursor(0, 0);
        lcd.print("STAR ");
        lcd.setCursor(13, 0);
        lcd.print('I');
        lcd.print(timeStarDelta / 1000);
        lcd.print('J');
        statusStarDelta = true;
        starDeltaMillis = millis();
        cDownMillis = millis();
    }
    else {
        statusStarDelta = false;
        //lcd.begin(16, 4);
        lcd.setCursor(0, 0);
        lcd.print(txtStartDelta);
        lcd.setCursor(13, 0);
        lcd.print('I');
        lcd.print(timeStarDelta / 1000);
        lcd.print('J');
    }
}
}
else if (digitalRead(UPpin) == LOW && digitalRead(DOWNpin) ==
LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {
        sdc = 0;
        lcd.begin(16, 4);
        lcd.clear();
        digitalWrite(StartSTARpin, HIGH);
        digitalWrite(StartDELTApin, HIGH);
        statusStarDelta = false;
        while (digitalRead(UPpin) == LOW && digitalRead(DOWNpin)
== LOW);
        tampilan();
    }
}

```



```

        arusMaxR = 0.0;
        arusMaxS = 0.0;
        arusMaxT = 0.0;
        break;
    }
}
else {
    cTombol = 0;
}
readArus();
displayValue();

if (statusStarDelta == true) {

    digitalWrite(StartSTARpin, LOW);
    if ( millis() - starDeltaMillis >= timeStarDelta && sdc <= 5) {
        digitalWrite(StartDELTApin, LOW);
        lcd.setCursor(5, 0);
        lcd.print("DELTA ");
    }
    //multitask
    if (millis() - cDownMillis >= 1000) {
        cDownMillis = millis();
        sdc++;
        if (sdc >= 25)sdc = 20;
        lcd.setCursor(0, 0);
        lcd.print("STAR ");

        if (sdc < 5) {
            lcd.setCursor(13, 0);
            lcd.print('[');
            lcd.print((timeStarDelta / 1000) - sdc);
            lcd.print(']');
            lcd.setCursor(15, 0);
        }
        else {
            lcd.setCursor(13, 0);
            lcd.print("RUN");
        }
    }
}

```

```

}

//blink panah
if (millis() - tombolMillis > 333) {
    tombolMillis = millis();
    sBlink = !sBlink;
    if (sBlink == true) {
        lcd.setCursor(11, 0);
        lcd.print(">>");
    }
    else if (sBlink == false) {
        lcd.setCursor(11, 0);
        lcd.print(" ");
    }
}
}
else {
    sdc = 0;
    digitalWrite(StartSTARpin, HIGH);
    digitalWrite(StartDELTApin, HIGH);

    if (millis() - cDownMillis >= 1000) {
        cDownMillis = millis();

        lcd.setCursor(0, 0);
        lcd.print(txtStartDelta);
        //lcd.setCursor(15, 0);
        //lcd.print(timeStarDelta );
    }
}
//blink kotak
if (millis() - tombolMillis > 333) {
    tombolMillis = millis();
    sBlink = !sBlink;
    if (sBlink == true) {
        lcd.setCursor(11, 0);
        lcd.print("<>");
    }
    else if (sBlink == false) {

```

```

        lcd.setCursor(11, 0);
        lcd.print(" ");
    }
}
}
}
}
}

```

```

long progressMillis = 0;
int cProgress = 0;

```

```

void modeSoftStart(void) {
    while (digitalRead(MENUpin) == LOW) {
        lcd.clear();
    }
    cProgress = 0;
    lcd.setCursor(0, 0);
    lcd.print(txtSoftStart);

```

```

while (1) {
    readArus();
    displayValue();

```

```

    if (digitalRead(MENUpin) == LOW) {
        cTombol++;
        if (cTombol >= dMax)cTombol = dLimit;
        if (cTombol == dValue) {
            lcd.begin(16, 4);
            lcd.setCursor(0, 0);
            lcd.print('I');
            lcd.print(cProgress);
            lcd.print("%]");

```

```

        if (softEnable == false) {
            softEnable = true;
            FlexiTimer2::start();
        }
        else if (softEnable == true) {

```

```

    softEnable = false;
    FlexiTimer2::stop();

    cProgress = 0;

    dutySet = 10;

    cDutyT = 0;
    dutyT = dutySet;
    gasspollT = false;

    cDutyS = 0;
    dutyS = dutySet;
    gasspollS = false;

    cDutyR = 0;
    dutyR = dutySet;
    gasspollR = false;

    digitalWrite(StartRSSpin, HIGH);
    digitalWrite(StartSSSpin, HIGH);
    digitalWrite(StartTSSpin, HIGH);

    lcd.setCursor(0, 0);
    lcd.print(txtSoftStart);
}
}
}
else if (digitalRead(UPpin) == LOW && digitalRead(DOWNpin) ==
LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {

        softEnable = false;
        FlexiTimer2::stop();

        dutySet = 10;

```

```

cDutyT = 0;
dutyT = dutySet;
gasspollT = false;

cDutyS = 0;
dutyS = dutySet;
gasspollS = false;

cDutyR = 0;
dutyR = dutySet;
gasspollR = false;

digitalWrite(StartRSSpin, HIGH);
digitalWrite(StartSSSpin, HIGH);
digitalWrite(StartTSSpin, HIGH);

digitalWrite(StartRSSpin, HIGH);
digitalWrite(StartSSSpin, HIGH);
digitalWrite(StartTSSpin, HIGH);

lcd.clear();
while (digitalRead(UPpin) == LOW && digitalRead(DOWNpin)
== LOW);
    tampilan();
    arusMaxR = 0.0;
    arusMaxS = 0.0;
    arusMaxT = 0.0;
    break;
}
}
else {
    cTombol = 0;
}

////////////////////////////////////

if (softEnable) {
    //Punya e T
    if (gasspollT == false) {

```

```

if (digitalRead(ZCTpin) == LOW) {
    gasspollT = true;
    softMillisT = millis();
}
}
if (dutyT > 0) {
    if (millis() - softMillisT > (lamaStart * 100)) {
        softMillisT = millis();
        dutyT--;
    }
}
//Punya e S
if (gasspollS == false) {
    if (digitalRead(ZCSpin) == LOW) {
        gasspollS = true;
        softMillisS = millis();
    }
}
if (dutyS > 0) {
    if (millis() - softMillisS > (lamaStart * 100)) {
        softMillisS = millis();
        dutyS--;
    }
}
//Punya e R
if (gasspollR == false) {
    if (digitalRead(ZCRpin) == LOW) {
        gasspollR = true;
        softMillisR = millis();
    }
}
if (dutyR > 0) {
    if (millis() - softMillisR > (lamaStart * 100)) {
        softMillisR = millis();
        dutyR--;
    }
}
//progres bar
if (millis() - progressMillis > (lamaStart * 10)) {

```

```

    progressMillis = millis();
    cProgress++;
    if (cProgress >= 100)cProgress = 100;
    lcd.setCursor(0, 0);
    lcd.print('I');

    lcd.print(cProgress);
    lcd.print("%");

    lcd.setCursor(5 + (cProgress/10), 0);
    lcd.print('#');
  }
}
else {
  dutySet = 10;
  gasspollR = false;
  gasspollS = false;
  gasspollT = false;
  digitalWrite(StartRSSpin, HIGH);
  digitalWrite(StartSSSpin, HIGH);
  digitalWrite(StartTSSpin, HIGH);
}
}
}
}

```

```

void modeSetSoftStart(void) {
  float lamaNYA=0.0;

  lamaNYA = lamaStart;
  while (digitalRead(MENUpin) == LOW) {
    lcd.clear();
  }
  lcd.setCursor(0, 0);
  lcd.print(txtSet);
  lcd.setCursor(0, 1);
  lcd.print("Time: ");
  lcd.print(lamaNYA,1);
  lcd.print("second ");
}

```

```

while (1) {

    if (digitalRead(UPpin) == LOW && digitalRead(DOWNpin) ==
LOW) {
        cTombol++;
        if (cTombol >= dMax)cTombol = dLimit;
        if (cTombol == dValue) {
            lcd.setCursor(0, 2);
            lcd.print("Use: ");
            lcd.print(lamaStart,1);
            lcd.print(" second ");
            delay(2500);
            lcd.clear();
            while (digitalRead(UPpin) == LOW && digitalRead(DOWNpin)
== LOW);
            break;
            tampilan();
        }
    }
    else if (digitalRead(MENUpin) == LOW) {
        cTombol++;
        if (cTombol >= dMax)cTombol = dLimit;
        if (cTombol == dValue) {
            lamaStart = lamaNYA;
            lcd.setCursor(0, 2);
            lcd.print("Set: ");
            lcd.print(lamaStart,1);
            lcd.print(" second ");
            delay(2500);
            lcd.clear();
            break;
            tampilan();
        }
    }
    else if (digitalRead(UPpin) == LOW) {
        cTombol++;
        if (cTombol >= dMax/2)cTombol = dLimit/2;

```



```

    if (cTombol == dValue/2) {
        lamaNYA=lamaNYA + 0.1;
        if (lamaNYA >= 11.0)lamaNYA = 11.0;
        lcd.setCursor(0, 0);
        lcd.print(txtSet);
        lcd.setCursor(0, 1);
        lcd.print("Time: ");
        lcd.print(lamaNYA,1);
        lcd.print("second ");
    }
}
else if (digitalRead(DOWNpin) == LOW) {
    cTombol++;
    if (cTombol >= dMax/2)cTombol = dLimit/2;
    if (cTombol == dValue/2) {
        lamaNYA = lamaNYA - 0.1;
        if (lamaNYA <= 4.0)lamaNYA = 4.0;
        lcd.setCursor(0, 0);
        lcd.print(txtSet);
        lcd.setCursor(0, 1);
        lcd.print("Time: ");
        lcd.print(lamaNYA,1);
        lcd.print("second ");
    }
}
else {
    cTombol = 0;
}

}
}

void readKey(void) {
    if (digitalRead(MENUpin) == LOW) {
        cTombol++;
        if (cTombol >= dMax)cTombol = dLimit;
        if (cTombol == dValue) {

            switch (baris){

```

```

    case 0:
        modeDOL();
        break;
    case 1:
        modeStarDelta();
        break;
    case 2:
        modeSoftStart();
        break;
    case 3:
        modeSetSoftStart();
        break;
    default:
        break;
}

}

}

else if (digitalRead(UPpin) == LOW && digitalRead(DOWNpin) ==
LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {
        atHome = true;
        lcd.clear();
    }
}

else if (digitalRead(UPpin) == LOW) {
    cTombol++;
    if (cTombol >= dMax)cTombol = dLimit;
    if (cTombol == dValue) {
        lcd.setCursor(0, baris);
        lcd.print(' ');

        baris--;
        if (baris < 0) baris = 3;
    }
}

else if (digitalRead(DOWNpin) == LOW) {

```

```

cTombol++;
if (cTombol >= dMax)cTombol = dLimit;
if (cTombol == dValue) {
    lcd.setCursor(0, baris);
    lcd.print(' ');

    baris++;
    if (baris >3) baris = 0;
}
}
else {
    cTombol = 0;
}
}
////blink kursor
if (millis() - tombolMillis > 333) {
    tombolMillis = millis();
    sBlink = !sBlink;
    if (sBlink == true) {
        lcd.setCursor(0, baris);
        lcd.print('>');
    }
    else if (sBlink == false) {
        lcd.setCursor(0, baris);
        lcd.print(' ');
    }
}
}
}

void displayHome(void) {
    lcd.setCursor(0, 0);
    lcd.print(" MODUL STARTING ");
    lcd.setCursor(0, 1);
    lcd.print("MOTOR IND 3 FASA");
    lcd.setCursor(0, 2);
    lcd.print(">> DANGEROUS! <<");
    lcd.setCursor(0, 3);
    lcd.print("--High Voltage--");

}

```

```

void tampilan(void)
{
    lcd.setCursor(1, 0);
    lcd.print(txtDOL);
    lcd.setCursor(1, 1);
    lcd.print(txtStartDelta);
    lcd.setCursor(1, 2);
    lcd.print(txtSoftStart);
    lcd.setCursor(1, 3);
    lcd.print(txtSet);
}

```

```

double akarDua = 1.414213562373095;
float faktorArus = 1.35;
long aMillis = 0;
int tempR=0, tempS=0, tempT=0;
int cSampling = 0;
int cArus = 0;

```

```

long rataMinR=0, rataMaxR=0, rataMinS = 0, rataMaxS = 0, rataMinT
= 0, rataMaxT = 0;

```

```

float nolAmpere = 0.06;

```

```

void readArus(void) {

    if (micros() - aMillis >= 1000) {
        aMillis = micros();
        //signed int AA = analogRead(CurrentRpin);
        //signed int BB = analogRead(CurrentSpin);
        //signed int CC = analogRead(CurrentTpin);
        int AA = read_adc(currentRpin);
        int BB = read_adc(currentSpin);
        int CC = read_adc(currentTpin);

        if (AA < minR)minR = AA;
        else if (AA > maxR)maxR = AA;
    }
}

```

```

if (BB < minS)minS = BB;
else if (BB > maxS)maxS = BB;

if (CC < minT)minT = CC;
else if (CC > maxT)maxT = CC;

cSampling++;

if (cSampling > 20) {
    cSampling = 0;

    rataMaxR += maxR;
    rataMinR += minR;
    maxR = 0;
    minR = 1023;

    rataMaxS += maxS;
    rataMinS += minS;
    maxS = 0;
    minS = 1023;

    rataMaxT += maxT;
    rataMinT += minT;
    maxT = 0;
    minT = 1023;

    cArus++;
}
if (cArus > 25) {
    cArus = 0;

    rataMaxR /= 25;
    rataMinR /= 25;
    float vppR = (rataMaxR - rataMinR) * 4.89 / 1023;
    rataMinR = 0;
    rataMaxR = 0;
    float vrmsR = vppR / 2 * akarDua;
    arusR = vrmsR-(nolAmpere + 0.03);

```

```

    arusR *= faktorArus;

    rataMaxS /= 25;
    rataMinS /= 25;
    float vppS = (rataMaxS - rataMinS) * 4.89 / 1023;
    rataMinR = 0;
    rataMaxR = 0;
    float vrmsS = vppS / 2 * akarDua;
    arusS = vrmsS - (nolAmpere + 0.03);
    arusS *= faktorArus;

    rataMaxT /= 25;
    rataMinT /= 25;
    float vppT = (rataMaxT - rataMinT) * 4.89 / 1023;
    rataMinT = 0;
    rataMaxT = 0;
    float vrmsT = vppT / 2 * akarDua;
    arusT = vrmsT - nolAmpere;
    arusT *= faktorArus;

    if (arusR < 0) arusR *= -1;
    if (arusS < 0) arusS *= -1;
    if (arusT < 0) arusT *= -1;

    if (arusR > arusMaxR) arusMaxR = arusR;
    if (arusS > arusMaxS) arusMaxS = arusS;
    if (arusT > arusMaxT) arusMaxT = arusT;

}

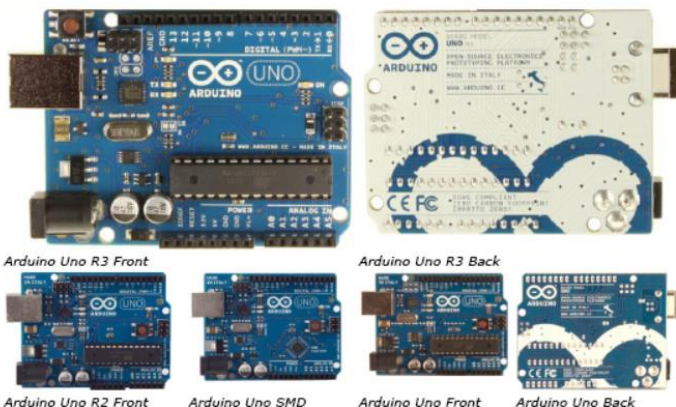
}

```

## LAMPIRAN B

### B.1 Datasheet Arduino Uno

#### Arduino Uno



#### Overview

The Arduino Uno is a microcontroller board based on the ATmega328 ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

[Revision 2](#) of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode.

[Revision 3](#) of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

#### Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V

Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

## Schematic & Reference Design

EAGLE files: [arduino-uno-Rev3-reference-design.zip](#) (NOTE: works with Eagle 6.0 and newer)

Schematic: [arduino-uno-Rev3-schematic.pdf](#)

**Note:** The Arduino reference design can use an ATmega8, 168, or 328. Current models use an ATmega328, but an ATmega8 is shown in the schematic for reference. The pin configuration is identical on all three processors.

## Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

## Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

## Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.



- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the [SPI library](#).
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the `analogReference()` function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the [Wire library](#).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with `analogReference()`.
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and ATmega328 ports](#). The mapping for the ATmega8, 168, and 328 is identical.

## Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, [on Windows, a .inf file is required](#). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. For SPI communication, use the [SPI library](#).

## Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See [this user-contributed tutorial](#) for more information.

## Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data. The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

## USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

## Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

## B.2 Datasheet PC817

**SHARP**

PC817 Series

# PC817 Series

## High Density Mounting Type Photocoupler

- Lead forming type (I type) and taping reel type (P type) are also available. (PC817/PC817P)
- TÜV (VDE0884) approved type is also available as an option.

### ■ Features

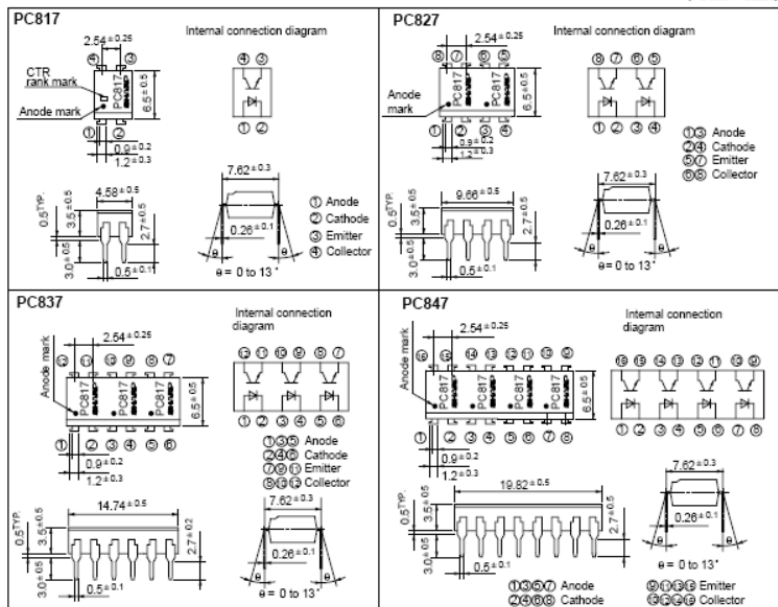
- Current transfer ratio  
(CTR: MIN. 50% at  $I_F = 5\text{mA}$ ,  $V_{CE} = 5\text{V}$ )
- High isolation voltage between input and output ( $V_{iso}$ : 5 000V<sub>rms</sub>)
- Compact dual-in-line package  
PC817: 1-channel type  
PC827: 2-channel type  
PC837: 3-channel type  
PC847: 4-channel type
- Recognized by UL, file No. E64380

### ■ Applications

- Computer terminals
- System appliances, measuring instruments
- Registers, copiers, automatic vending machines
- Electric home appliances, such as fan heaters, etc.
- Signal transmission between circuits of different potentials and impedances

### ■ Outline Dimensions

(Unit: mm)



\* In the absence of confirmation by device specification sheets, SHARP takes no responsibility for any defects that occur in equipment using any of SHARP's devices, shown in catalogs, data books, etc. Contact SHARP in order to obtain the latest version of the device specification sheets before using any SHARP's device.\*

## ■ Absolute Maximum Ratings

(Ta = 25°C)

Parameter	Symbol	Rating	Unit
Forward current	$I_F$	50	mA
*1 Peak forward current	$I_{FM}$	1	A
Reverse voltage	$V_R$	6	V
Power dissipation	$P$	70	mW
Collector-emitter voltage	$V_{CEO}$	35	V
Emitter-collector voltage	$V_{ECO}$	6	V
Collector current	$I_C$	50	mA
Collector power dissipation	$P_C$	150	mW
Total power dissipation	$P_{tot}$	200	mW
*2 Isolation voltage	$V_{iso}$	5 000	V <sub>rms</sub>
Operating temperature	$T_{op}$	-30 to +100	°C
Storage temperature	$T_{stg}$	-55 to +125	°C
*3 Soldering temperature	$T_{sld}$	260	°C

\*1 Pulse width:  $\leq 100\mu s$ , Duty ratio: 0.001

\*2 40 to 60% RH, AC for 1 minute

\*3 For 10 seconds

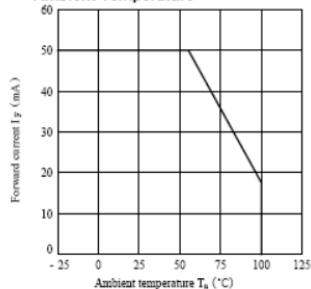
## ■ Electro-optical Characteristics

(Ta = 25°C)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Forward voltage	$V_F$	$I_F = 20\text{mA}$	-	1.2	1.4	V
Peak forward voltage	$V_{FM}$	$I_{FM} = 0.5\text{A}$	-	-	3.0	V
Reverse current	$I_R$	$V_R = 4\text{V}$	-	-	10	$\mu\text{A}$
Terminal capacitance	$C_t$	$V = 0, f = 1\text{kHz}$	-	30	250	pF
Collector dark current	$I_{CBO}$	$V_{CE} = 20\text{V}$	-	-	$10^{-7}$	A
*4 Current transfer ratio	CTR	$I_F = 5\text{mA}, V_{CE} = 5\text{V}$	50	-	600	%
Collector-emitter saturation voltage	$V_{CE(sat)}$	$I_F = 20\text{mA}, I_C = 1\text{mA}$	-	0.1	0.2	V
Isolation resistance	$R_{iso}$	DC500V, 40 to 60% RH	$5 \times 10^{10}$	$10^{11}$	-	$\Omega$
Floating capacitance	$C_f$	$V = 0, f = 1\text{MHz}$	-	0.6	1.0	pF
Cut-off frequency	$f_c$	$V_{CE} = 5\text{V}, I_C = 2\text{mA}, R_L = 100\Omega, -3\text{dB}$	-	80	-	kHz
Response time	Rise time	$V_{CE} = 2\text{V}, I_C = 2\text{mA}, R_L = 100\Omega$	-	4	18	$\mu s$
	Fall time		-	3	18	$\mu s$

\*4 Classification table of current transfer ratio is shown below.

Fig. 1 Forward Current vs. Ambient Temperature



Model No.	Rank mark	CTR (%)
PC817A	A	80 to 160
PC817B	B	130 to 260
PC817C	C	200 to 400
PC817D	D	300 to 600
PC8●7AB	A or B	80 to 260
PC8●7BC	B or C	130 to 400
PC8●7CD	C or D	200 to 600
PC8●7AC	A, B or C	80 to 400
PC8●7BD	B, C or D	130 to 600
PC8●7AD	A, B, C or D	80 to 600
PC8●7	A, B, C, D or No mark	50 to 600

● : 1 or 2 or 3 or 4

Fig. 2 Collector Power Dissipation vs. Ambient Temperature

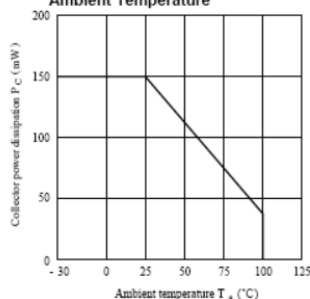


Fig. 3 Peak Forward Current vs. Duty Ratio

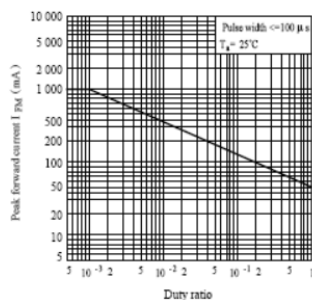


Fig. 4 Current Transfer Ratio vs. Forward Current

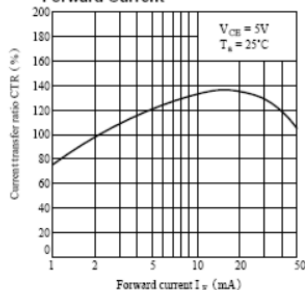


Fig. 5 Forward Current vs. Forward Voltage

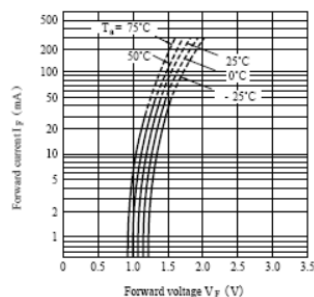


Fig. 6 Collector Current vs. Collector-emitter Voltage

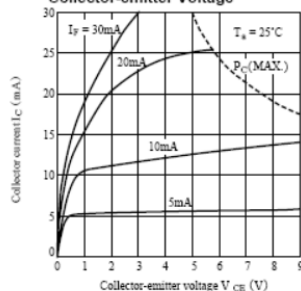


Fig. 7 Relative Current Transfer Ratio vs. Ambient Temperature

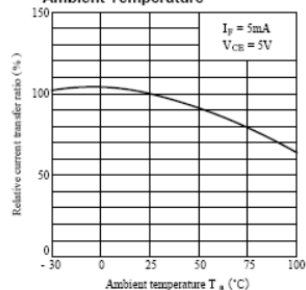


Fig. 8 Collector-emitter Saturation Voltage vs. Ambient Temperature

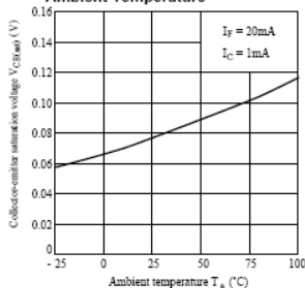


Fig. 9 Collector Dark Current vs. Ambient Temperature

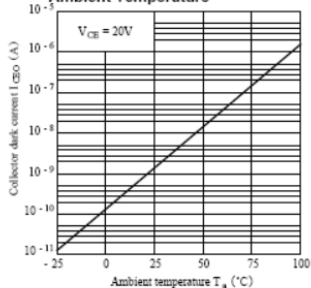


Fig. 10 Response Time vs. Load Resistance

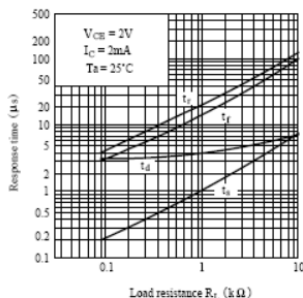
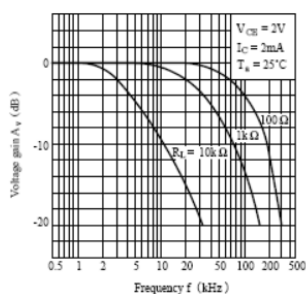
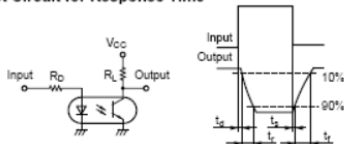


Fig. 11 Frequency Response



Test Circuit for Response Time



Test Circuit for Frequency Response

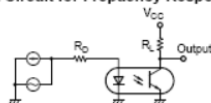
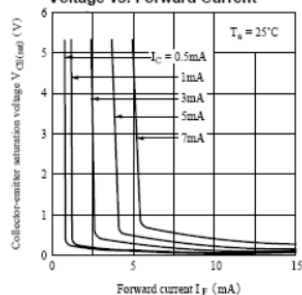


Fig. 12 Collector-emitter Saturation Voltage vs. Forward Current



● Please refer to the chapter "Precautions for Use"

## B.3 Datasheet MOC3021

### MOTOROLA SEMICONDUCTOR TECHNICAL DATA

Order this document  
by MOC3020/D



## 6-Pin DIP Random-Phase Optoisolators Triac Driver Output (400 Volts Peak)

The MOC3020 Series consists of gallium arsenide infrared emitting diodes, optically coupled to a silicon bilateral switch.

- To order devices that are tested and marked per VDE 0884 requirements, the suffix "V" must be included at end of part number. VDE 0884 is a test option. They are designed for applications requiring isolated triac triggering.

Recommended for 115/240 Vac(rms) Applications:

- Solenoid/Valve Controls
- Lamp Ballasts
- Interfacing Microprocessors to 115 Vac Peripherals
- Motor Controls
- Static ac Power Switch
- Solid State Relays
- Incandescent Lamp Dimmers

MAXIMUM RATINGS ( $T_A = 25^\circ\text{C}$  unless otherwise noted)

Rating	Symbol	Value	Unit
<b>INFRARED EMITTING DIODE</b>			
Reverse Voltage	$V_R$	3	Volts
Forward Current — Continuous	$I_F$	60	mA
Total Power Dissipation @ $T_A = 25^\circ\text{C}$ Negligible Power in Triac Driver Derate above $25^\circ\text{C}$	$P_D$	100 1.33	mW mW/°C

#### OUTPUT DRIVER

Off-State Output Terminal Voltage	$V_{DRM}$	400	Volts
Peak Repetitive Surge Current (PW = 1 ms, 120 pps)	$I_{TSM}$	1	A
Total Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above $25^\circ\text{C}$	$P_D$	300 4	mW mW/°C

#### TOTAL DEVICE

Isolation Surge Voltage <sup>(1)</sup> (Peak ac Voltage, 60 Hz, 1 Second Duration)	$V_{ISO}$	7500	Vac(pk)
Total Power Dissipation @ $T_A = 25^\circ\text{C}$ Derate above $25^\circ\text{C}$	$P_D$	330 4.4	mW mW/°C
Junction Temperature Range	$T_J$	-40 to +100	°C
Ambient Operating Temperature Range <sup>(2)</sup>	$T_A$	-40 to +85	°C
Storage Temperature Range <sup>(2)</sup>	$T_{stg}$	-40 to +150	°C
Soldering Temperature (10 s)	$T_L$	260	°C

- Isolation surge voltage,  $V_{ISO}$ , is an internal device dielectric breakdown rating. For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.
- Refer to Quality and Reliability Section in Opto Data Book for information on test conditions.

Preferred devices are Motorola recommended choices for future use and best overall value.  
Global Optoisolator is a trademark of Motorola, Inc.

REV 1

© Motorola, Inc. 1995

**MOC3021**  
(IFT = 15 mA Max)  
**MOC3022**  
(IFT = 10 mA Max)  
**MOC3023\***  
(IFT = 5 mA Max)

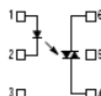
\*Motorola Preferred Device

STYLE 6 PLASTIC



STANDARD THRU HOLE  
CASE 730A-04

SCHEMATIC



- ANODE
- CATHODE
- NC
- MAIN TERMINAL
- SUBSTRATE  
DO NOT CONNECT
- MAIN TERMINAL



**MOTOROLA**

# MOC3021 MOC3022 MOC3023

ELECTRICAL CHARACTERISTICS ( $T_A = 25^\circ\text{C}$  unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
<b>INPUT LED</b>					
Reverse Leakage Current ( $V_R = 3\text{ V}$ )	$I_R$	—	0.05	100	$\mu\text{A}$
Forward Voltage ( $I_F = 10\text{ mA}$ )	$V_F$	—	1.15	1.5	Volts
<b>OUTPUT DETECTOR (<math>I_F = 0</math> unless otherwise noted)</b>					
Peak Blocking Current, Either Direction (Rated $V_{DRM}^{(1)}$ )	$I_{DRM}$	—	10	100	nA
Peak On-State Voltage, Either Direction ( $I_{TM} = 100\text{ mA Peak}$ )	$V_{TM}$	—	1.8	3	Volts
Critical Rate of Rise of Off-State Voltage (Figure 7, Note 2)	$dv/dt$	—	10	—	$\text{V}/\mu\text{s}$
<b>COUPLED</b>					
LED Trigger Current, Current Required to Latch Output (Main Terminal Voltage = $3\text{ V}^{(3)}$ )	$I_{FT}$	—	8	15	mA
MOC3021	—	—	—	10	
MOC3022	—	—	—	5	
MOC3023	—	—	—	—	
Holding Current, Either Direction	$I_H$	—	100	—	$\mu\text{A}$

- Test voltage must be applied within  $dv/dt$  rating.
- This is static  $dv/dt$ . See Figure 7 for test circuit. Commutating  $dv/dt$  is a function of the load-driving thyristor(s) only.
- All devices are guaranteed to trigger at an  $I_F$  value less than or equal to max  $I_{FT}$ . Therefore, recommended operating  $I_F$  lies between max  $I_{FT}$  (15 mA for MOC3021, 10 mA for MOC3022, 5 mA for MOC3023) and absolute max  $I_F$  (60 mA).

## TYPICAL ELECTRICAL CHARACTERISTICS

$T_A = 25^\circ\text{C}$

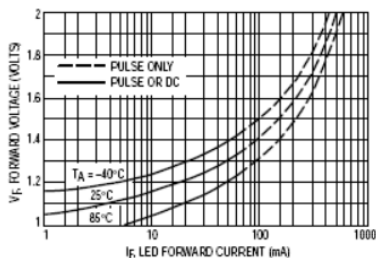


Figure 1. LED Forward Voltage versus Forward Current

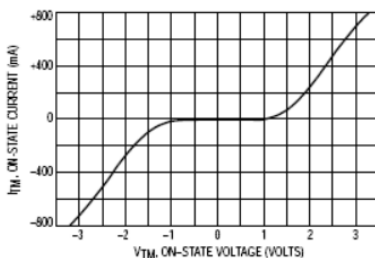


Figure 2. On-State Characteristics



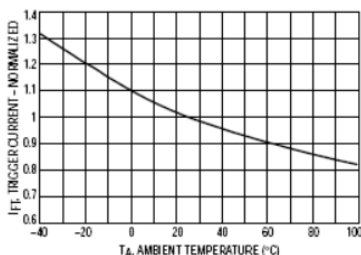


Figure 3. Trigger Current versus Temperature

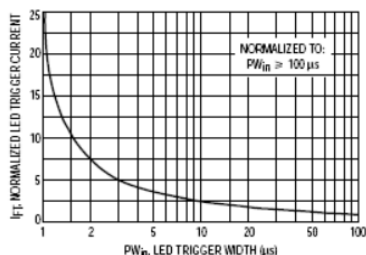


Figure 4. LED Current Required to Trigger versus LED Pulse Width

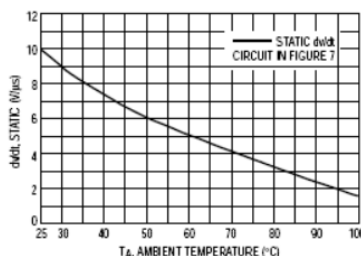


Figure 5. dv/dt versus Temperature

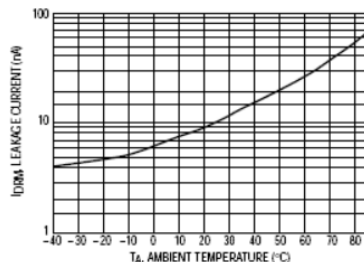


Figure 6. Leakage Current, IDRM versus Temperature

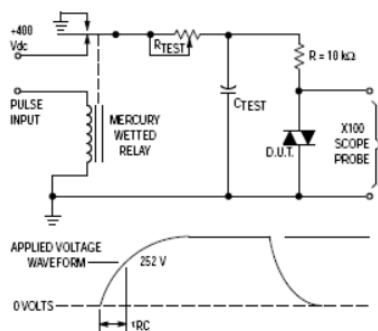
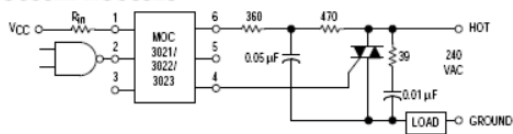


Figure 7. Static dv/dt Test Circuit

1. The mercury wetted relay provides a high speed repeated pulse to the D.U.T.
2. 100x scope probes are used, to allow high speeds and voltages.
3. The worst-case condition for static dv/dt is established by triggering the D.U.T. with a normal LED input current, then removing the current. The variable  $R_{TEST}$  allows the dv/dt to be gradually increased until the D.U.T. continues to trigger in response to the applied voltage pulse, even after the LED current has been removed. The dv/dt is then decreased until the D.U.T. stops triggering.  $\tau_{RC}$  is measured at this point and recorded.

# MOC3021 MOC3022 MOC3023



\* This optoisolator should not be used to drive a load directly. It is intended to be a trigger device only.

Additional information on the use of optically coupled triac drivers is available in Application Note AN-780A.

In this circuit the "hot" side of the line is switched and the load connected to the cold or ground side.

The 39 ohm resistor and 0.01  $\mu$ F capacitor are for snubbing of the triac, and the 470 ohm resistor and 0.05  $\mu$ F capacitor are for snubbing the coupler. These components may or may not be necessary depending upon the particular triac and load used.

Figure 8. Typical Application Circuit

## B.4 Datasheet BT138

NXP Semiconductors

Product specification

### Triacs

### BT138 series

#### GENERAL DESCRIPTION

Passivated triacs in a plastic envelope, intended for use in applications requiring high bidirectional transient and blocking voltage capability and high thermal cycling performance. Typical applications include motor control, industrial and domestic lighting, heating and static switching.

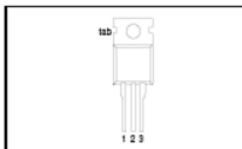
#### QUICK REFERENCE DATA

SYMBOL	PARAMETER	MAX.	MAX.	UNIT
$V_{ORM}$	BT138-600F	600	800	V
	BT138-600G	600F	800F	
	BT138-600G	600G	800G	
	600	600	800	
$I_{T(RMS)}$	RMS on-state current	12	12	A
$I_{TSM}$	Non-repetitive peak on-state current	95	95	A

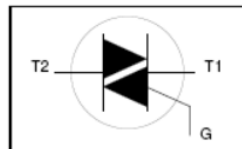
#### PINNING - TO220AB

PIN	DESCRIPTION
1	main terminal 1
2	main terminal 2
3	gate
tab	main terminal 2

#### PIN CONFIGURATION



#### SYMBOL



#### LIMITING VALUES

Limiting values in accordance with the Absolute Maximum System (IEC 134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.		UNIT
$V_{ORM}$	Repetitive peak off-state voltages		-	-600 600 <sup>1</sup>	-800 800	V
$I_{T(RMS)}$	RMS on-state current	full sine wave; $T_{MC} \leq 99^\circ\text{C}$	-	12		A
$I_{TSM}$	Non-repetitive peak on-state current	full sine wave; $T_J = 25^\circ\text{C}$ prior to surge	-	95		A
$I_{ft}$	$I_{ft}$ for fusing	$t = 20\text{ ms}$	-	105		A
$di_T/dt$	Repetitive rate of rise of on-state current after triggering	$t = 10\text{ ms}$	-	45		A/s
		$I_{TM} = 20\text{ A}$ ; $I_G = 0.2\text{ A}$ ; $di_G/dt = 0.2\text{ A}/\mu\text{s}$				
$I_{GM}$	Peak gate current	T2+ G+	-	50		A/ $\mu\text{s}$
$V_{GM}$	Peak gate voltage	T2+ G-	-	50		A/ $\mu\text{s}$
$P_{GM}$	Peak gate power	T2- G-	-	50		A/ $\mu\text{s}$
$P_{GM(AV)}$	Average gate power	T2- G+	-	10		A/ $\mu\text{s}$
$T_{JG}$	Storage temperature		-	2		A
$T_J$	Operating junction temperature		-	5		V
			-	5		W
		over any 20 ms period	-	0.5		W
			-40	150		$^\circ\text{C}$
			-	125		$^\circ\text{C}$

<sup>1</sup> Although not recommended, off-state voltages up to 800V may be applied without damage, but the triac may switch to the on-state. The rate of rise of current should not exceed 15 A/ $\mu\text{s}$ .

## Triacs

## BT138 series

## THERMAL RESISTANCES

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
$R_{th-jb}$	Thermal resistance junction to mounting base	full cycle	-	-	1.5	K/W
$R_{th-ja}$	Thermal resistance junction to ambient	half cycle in free air	-	60	2.0	K/W

## STATIC CHARACTERISTICS

 $T_j = 25\text{ }^{\circ}\text{C}$  unless otherwise stated

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.			UNIT
$I_{GT}$	Gate trigger current	BT138- $V_D = 12\text{ V}; I_T = 0.1\text{ A}$			...	...F	...G	
		T2+ G+	-	5	35	25	50	mA
		T2+ G-	-	8	35	25	50	mA
		T2- G-	-	10	35	25	50	mA
		T2- G+	-	22	70	70	100	mA
$I_L$	Latching current	$V_D = 12\text{ V}; I_{GT} = 0.1\text{ A}$						
		T2+ G+	-	7	40	40	60	mA
		T2+ G-	-	20	60	60	90	mA
		T2- G-	-	8	40	40	60	mA
		T2- G+	-	10	60	60	90	mA
$I_H$	Holding current	$V_D = 12\text{ V}; I_{GT} = 0.1\text{ A}$	-	6	30	30	60	mA
$V_T$	On-state voltage	$I_T = 15\text{ A}$	-	1.4	1.65			V
$V_{GT}$	Gate trigger voltage	$V_D = 12\text{ V}; I_T = 0.1\text{ A}$	-	0.7	1.5			V
		$V_D = 400\text{ V}; I_T = 0.1\text{ A};$ $T_j = 125\text{ }^{\circ}\text{C}$	0.25	0.4	-			V
$I_D$	Off-state leakage current	$V_D = V_{DRM(max)};$ $T_j = 125\text{ }^{\circ}\text{C}$	-	0.1	0.5			mA

## DYNAMIC CHARACTERISTICS

 $T_j = 25\text{ }^{\circ}\text{C}$  unless otherwise stated

SYMBOL	PARAMETER	CONDITIONS	MIN.			TYP.	MAX.	UNIT
$dV_D/dt$	Critical rate of rise of off-state voltage	BT138- $V_{DM} = 67\% V_{DRM(max)};$ $T_j = 125\text{ }^{\circ}\text{C};$ exponential waveform; gate open circuit	...	...F	...G			
			100	50	200	250	-	V/ $\mu\text{s}$
$dV_{com}/dt$	Critical rate of change of commutating voltage	$V_{DM} = 400\text{ V}; T_j = 95\text{ }^{\circ}\text{C};$ $I_{T(RMS)} = 12\text{ A};$ $di_{com}/dt = 5.4\text{ A/ms};$ gate open circuit	-	-	10	20	-	V/ $\mu\text{s}$
$t_{gt}$	Gate controlled turn-on time	$I_{TM} = 16\text{ A}; V_D = V_{DRM(max)};$ $I_G = 0.1\text{ A}; di_G/dt = 5\text{ A}/\mu\text{s}$	-	-	-	2	-	$\mu\text{s}$

## Triacs

## BT138 series

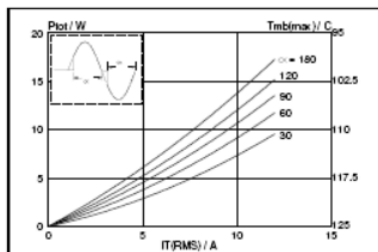


Fig.1. Maximum on-state dissipation,  $P_{om}$ , versus rms on-state current,  $I_{T(RMS)}$ , where  $\alpha$  = conduction angle.

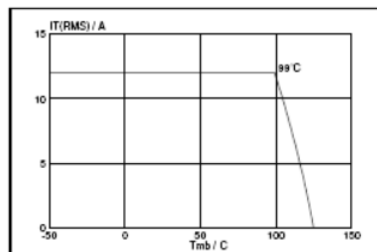


Fig.4. Maximum permissible rms current  $I_{T(RMS)}$ , versus mounting base temperature  $T_{mb}$ .

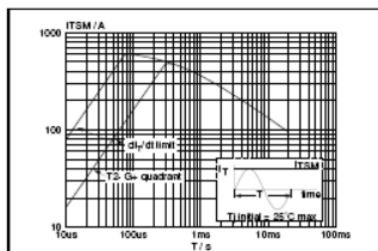


Fig.2. Maximum permissible non-repetitive peak on-state current  $I_{TSM}$  versus pulse width  $t_p$  for sinusoidal currents,  $t_c \leq 20\text{ms}$ .

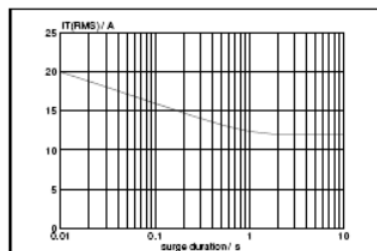


Fig.5. Maximum permissible repetitive rms on-state current  $I_{T(RMS)}$ , versus surge duration, for sinusoidal currents,  $f = 50\text{ Hz}$ ;  $T_{amb} \leq 90^\circ\text{C}$ .

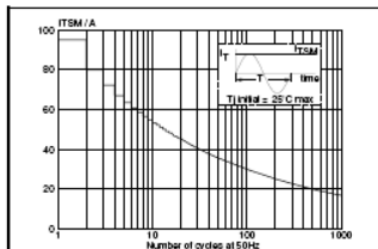


Fig.3. Maximum permissible non-repetitive peak on-state current  $I_{TSM}$  versus number of cycles, for sinusoidal currents,  $f = 50\text{ Hz}$ .

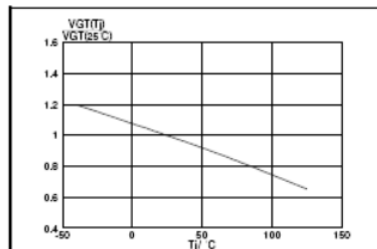


Fig.6. Normalised gate trigger voltage  $V_{GT}(T_j)/V_{GT}(25^\circ\text{C})$ , versus junction temperature  $T_j$ .

Triacs

BT138 series

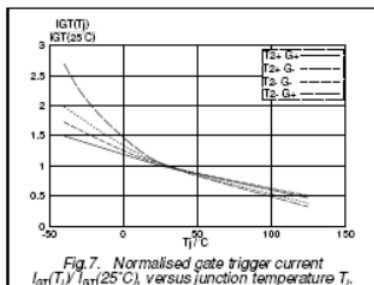


Fig.7. Normalised gate trigger current  $I_{GT}(T_J)/I_{GT}(25^\circ\text{C})$ , versus junction temperature  $T_J$ .

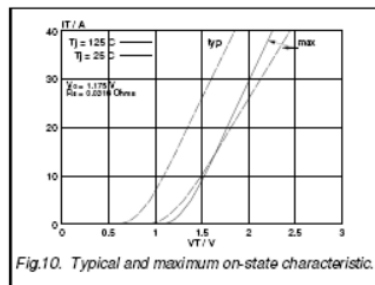


Fig.10. Typical and maximum on-state characteristic.

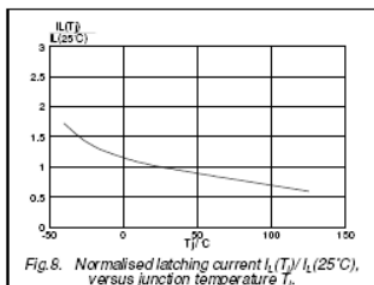


Fig.8. Normalised latching current  $I_L(T_J)/I_L(25^\circ\text{C})$ , versus junction temperature  $T_J$ .

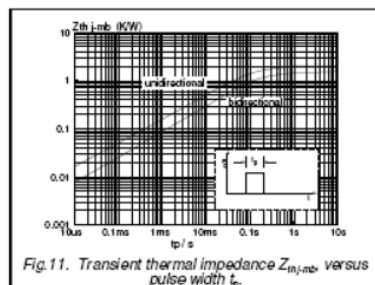


Fig.11. Transient thermal impedance  $Z_{thJA}$ , versus pulse width  $t_p$ .

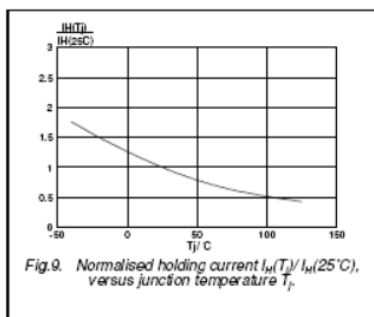


Fig.9. Normalised holding current  $I_H(T_J)/I_H(25^\circ\text{C})$ , versus junction temperature  $T_J$ .

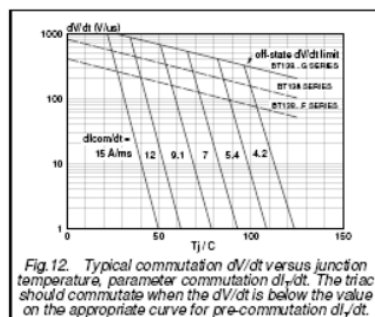


Fig.12. Typical commutation  $dV/dt$  versus junction temperature, parameter commutation  $dI/dt$ . The triac should commute when the  $dV/dt$  is below the value on the appropriate curve for pre-commutation  $dI/dt$ .

## B.5 Datasheet CT103

### Mini current transformer CT103



#### Features:

- 1- High accuracy, AC measurement accuracy up to class 1.0;
- 2- Small phase error, typical phase shift is at 1-10'; Additional compensation not needed;
- 3- Good isolation from Primary to Secondary;
- 4- Very reliable and stable performance at different temperature conditions;
- 5- Frequency: 50Hz/60Hz Ambient Temperature: -40°C ~ 70°C;

#### Specifications:

- Measurement Current: 100A
- Ratio: Current: 100A/50mA  
Turns: 2000
- Load: 12.5Ω
- Insulation Voltage: 3000V
- Phase Error: <15'
- Accuracy Class: 0.1

*Made in China*

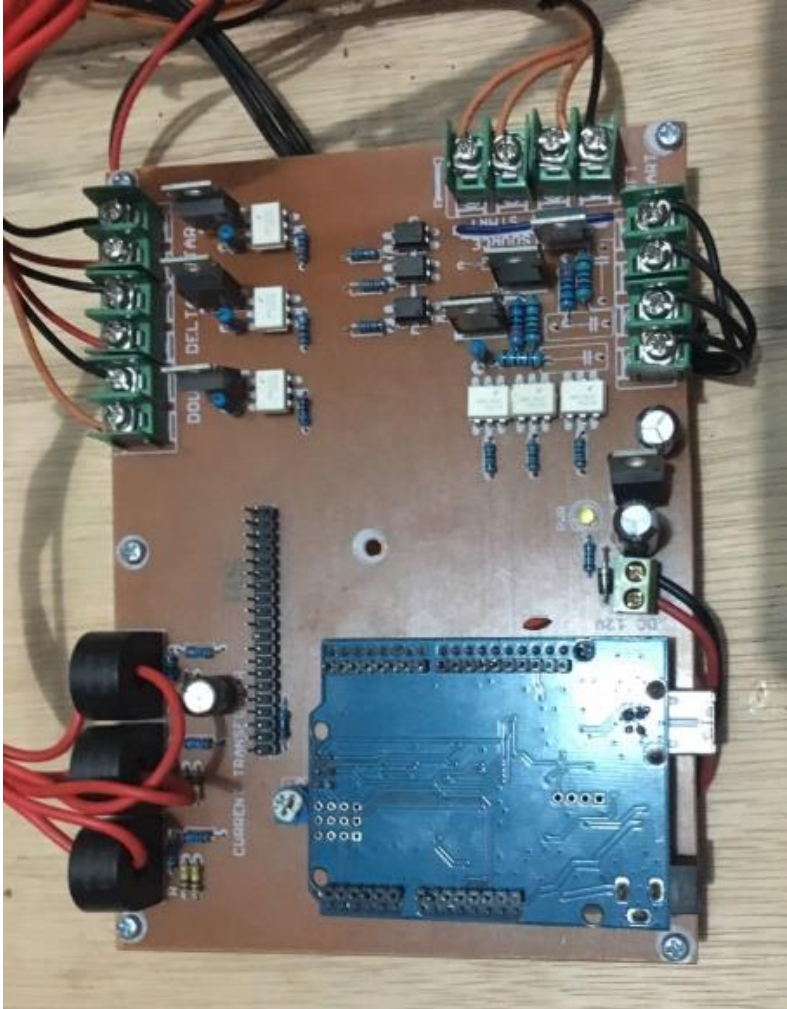
**EKT**  
Electronics K&B ang Trading

Halaman ini sengaja dikosongi

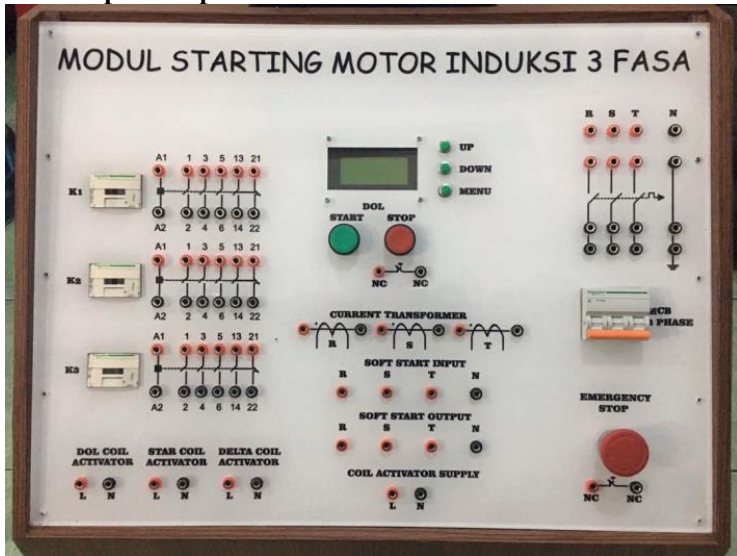


## LAMPIRAN C

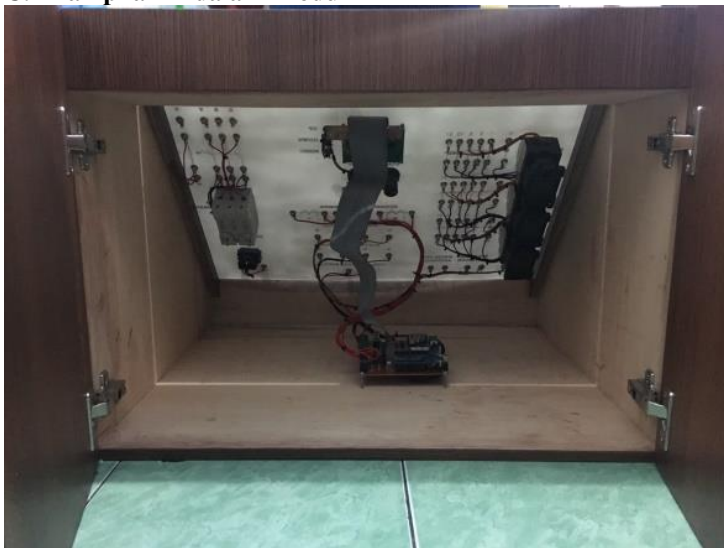
### C.1 Tampilan Board



## C.2 Tampilan Depan Modul



## C.2 Tampilan Didalam Modul



## **DAFTAR RIWAYAT HIDUP**



Nama : Yoki Rijal Fauzi  
TTL : Sukoharjo, 24 Desember 1995  
Jenis Kelamin : Laki - laki  
Agama : Islam  
Alamat : Begajah RT 2 RW 4, Kabupaten Sukoharjo  
Telp/HP : 081357223669  
E-mail : yokirf@gmail.com

### **RIWAYAT PENDIDIKAN**

1. 2002 – 2008 : SDN Gayam 1 Sukoharjo
2. 2008 – 2011 : SMP Negeri 1 Sukoharjo
3. 2011 – 2014 : SMA Negeri 1 Sukoharjo
4. 2014 – 2017 : D3 Teknik Elektro Otomasi, Program Studi Teknik Listrik – Fakultas Vokasi Institut Teknologi Sepuluh Nopember (ITS)

### **PENGALAMAN KERJA**

1. Kerja Praktek di PT PLN (Persero) APD Jawa Timur Area Surabaya Selatan

### **PENGALAMAN ORGANISASI**

1. Remaja Masjid Luhur Al-Ikhlas, Medokan Semampir.